DEPARTMENT OF MECHANICAL ENGINEERING & MECHANICS
COLLEGE OF ENGINEERING & TECHNOLOGY
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23529

CONVERGENCE ACCELERATION OF THE PROTEUS COMPUTER CODE WITH MULTIGRID METHODS

By

A. O. Demuren, Principal Investigator

and

S. O. Ibraheem, Graduate Research Assistant

Interim Report I (Progress Report)
For the period ended July 31, 1992

Prepared for
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, OH 44135

Under
Research Grant NAG-3-1329
J. R. Scott, Technical Monitor
Inlet, Duct, Nozzle Flow Physics Branch

August 1992

DEPARTMENT OF MECHANICAL ENGINEERING & MECHANICS
COLLEGE OF ENGINEERING & TECHNOLOGY
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23529

CONVERGENCE ACCELERATION OF THE PROTEUS COMPUTER CODE WITH MULTIGRID METHODS

By

A. O. Demuren, Principal Investigator

and

S. O. Ibraheem, Graduate Research Assistant

Interim Report I (Progress Report)
For the period ended July 31, 1992

Prepared for
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, OH 44135

Under
Research Grant NAG-3-1329
J. R. Scott, Technical Monitor
Inlet, Duct, Nozzle Flow Physics Branch

Submitted by the
Old Dominion University Research Foundation
P.O. Box 6369
Norfolk, Virginia 23508-0369

August 1992

SUMMARY

This report presents the first part of a study to implement convergence acceleration techniques based on the multigrid concept in the Proteus computer code. The report is in three sections. The first section presents a review of previous studies on the implementation of multigrid methods in computer codes for compressible flow analysis. The second section presents detailed stability analysis of upwind and central-difference based numerical schemes for solving the Euler and Navier-Stokes equations. The last section presents results of a convergence study of the Proteus code on computational grids of different sizes.

The results presented in this report form the foundation for the implementation of multigrid methods in the Proteus code which constitutes the second part of this study.

I : Review of Multigrid Methods for Compressible Flow Computation

Multiple grids were first proposed in the form of two-grid level schemes to accelerate the convergence of iterative procedures by Southwell (1935), Stiefel (1952), Federenko (1961), amongst others. Full multiple grid methods were introduced for the Poisson equation by Federenko (1964) and the approach was generalized by Bakhalov (1966) to any second-order elliptic operator with continuous coefficients. Perhaps the most influential work on the application of multigrid methods to elliptic type problems is the paper by Brandt (1977) which also introduced the use of local mode analysis to determine the smoothing rates of multigrid schemes. Multigrid acceleration was also successfully applied to the transonic potential flow equation, which is of mixed elliptic-hyperbolic type, by South and Brandt (1976), Jameson (1979), McCarthy and Reyhner (1982), etc..

Most of the theory of the effectiveness of multigrid schemes pertained to problems with some measure of ellipticity. However, Ni (1981) proposed a distributed correction multigrid method based on an explicit scheme for solving the Euler equations in the steady state. Convergence acceleration due to the multigrid scheme was by at most a factor of 5 which was worse than typical speedup factors in applications to elliptic equations. Furthermore the scheme was only first-order accurate and was restricted to a CFL number of one. Jameson (1983) proposed an explicit four-stage time stepping multigrid algorithm for the steady-state Euler equations. The method was second-order accurate and the limiting CFL number for stability was 2.6–2.8. The mechanism for multigrid convergence acceleration to steady state in systems with little ellipticity is that larger time steps can be taken on coarser grids, while still maintaining the same CFL number, such that disturbances are more rapidly expelled through the boundaries. The interpolation of corrections from the coarse grid to the fine grid may introduce additional

high frequency errors which must be rapidly damped if the scheme is to be effective. Thus a requirement of any solution scheme to be used successfully in a multigrid procedure is that it rapidly dampens high frequency modes of the error.

Mulder presented a multigrid scheme to solve the two dimensional Euler equation with a finite-volume method which used van Leer's flux-vector splitting for upwind differencing and a symmetric Gauss-Siedel method as a relaxation scheme. Multigrid speedup factors were roughly 9 and 6 for first-order and second-order accurate schemes, respectively. Anderson et al. (1988) also found similar multigrid convergence acceleration rates in the solution of three-dimensional Euler equations with flux-vector splitting and three different approximate factorization schemes. Typically, 200–400 multigrid cycles were required for convergence to the level of the truncation errors. An interesting result was that although the three-factor spatially split factorization was stable only for CFL numbers below 20, it produced the fastest multigrid convergence of all the schemes. This was obtained at a CFL number of 7.

Jameson and Yoon (1986) presented finite-volume based multigrid methods for the 2–D Euler equations using an ADI scheme with approximate factorization. The differential operators were approximated with central differences with second and fourth-difference artificial dissipation terms added for stability and convergence. It was found that implicit fourth difference dissipation was required for efficient multigrid convergence . However, this required the solution of a block pentadiagonal system which was more expensive than the block tridiagonal system resulting from the use of implicit second-difference dissipation. Compared with the single grid computation the multigrid speedup factor (based on residual reduction) was about 8 in the former and 4 in the latter. Multigrid methods coupled with grid sequencing enabled quite rapid establishment of the solution

fields, so that based on the buildup of the supersonic region, the speedup factors in the study above were twice as large. The problem with the ADI scheme as a baseline solver is that in three-dimensions, a three-factor split is required and linearized stability analysis shows that this is only conditionally stable. To alleviate this problem Jameson and Yoon (1987) devised a multigrid method for 2-D Euler equations which used the lower-upper (LU) factorized implicit scheme of Jameson and Turkel (1981) as the baseline solver. Yokota and Caughey (1988) have developed a similar scheme for the calculation of three-dimensional transonic flow through rotating cascades. The scheme has only two factors and is unconditionally stable. It is indeed very similar to the flux-vector splitting method based on the eigenvalue factored split investigated by Anderson et al. (1988). Their finding that although the three-factored split (similar to ADI) is only conditionally stable, it provided a better multigrid convergence rate than the unconditionally stable eigenvalue split method (similar to LU), is noteworthy. However, one advantage of the LU scheme is that it requires cheaper block-bidiagonal inversions compared with block-tridiagonal or pentadiagonal inversions for an ADI scheme. The latter is necessary if implicit fourth-difference dissipation terms are used for better accuracy and convergence. Caughey (1988) demonstrated that block- pentadiagonal inversions in the ADI scheme could be reduced to scalar pentadiagonal ones by using a local similarity transformation to diagonalize the equations at each point. Thus, the computational work was reduced by a factor of four, and the decoupled system had similar convergence characteristics as the original one. Caughey and Iyer (1989) applied the scheme to solve the Euler equations for a supersonic inlet flow and found that the multigrid speedup factor was only 2.5, i.e., somewhat less than was found in transonic and subsonic flows. Yokota, Caughey and Chima (1988) also diagonalized the LU implicit multigrid scheme with no

degradation in performance.

So far in this review, we have considered the application of multigrid methods to the Euler equations or potential flow equations. Several applications to Navier-Stokes equations for incompressible fluid flow have been reported (Vanka 1986, Philips et al. 1987, Demuren 1989, Thompson and Ferziger 1989, Demuren 1992). The relaxation schemes in all these applications are pressure-based in contrast to time-stepping schemes more common in compressible flow applications. Multigrid speedup in the range of a few percent to factors of hundreds have been reported. It is likely that in the latter cases, the baseline relaxation scheme did not have good convergence properties for the particular applications. However, one of the attractions of the multigrid method is that a poor single-grid solver may actually have good high frequency smoothing properties and thus be an effective multigrid relaxation scheme. Rhie (1989) presented a pressure-based multigrid method for solving the Navier-Stokes equations over the range of flow speeds encompassing both the compressible and the incompressible fluid flow. Himansu and Rubin (1988) also presented a novel pressure-based multigrid method for the reduced Navier-Stokes equations for compressible and incompressible fluid flows. Apart from the obvious difficulties of the treatment of viscous terms and the implementation of a turbulence model, the solution of Navier-Stokes equations usually requires the clustering of grids near walls in order to resolve the boundary layer, which often increases the stiffness of the system of equations and slows down the convergence rate of many iterative schemes. Multigrid convergence acceleration also tended to degrade with increase in Reynolds number. These difficulties fall under the category of problems with standard multigrid methods classified by Brandt (1977) as due to the alignment of coefficients of difference equations. He proposed to overcome the problem by doing line relaxations in

2-D or plane relaxations in 3-D in the direction of alignment, or to perform only semi-coarsenig of the grids in one of the directions instead of the more usual full coarsening, which should reduce the anisotropy of the coefficients. Himansu and Rubin (1988) implemented some aspect of both strategies with some success. Mulder (1989) considered the problem of alignment in somewhat more details and found that semi-coarsening in one direction was inadequate to cure it. Rather, it must be used in several directions at every grid level. Hence, in a 2-D problem two coarse grids are created for each finer grid, which implies that the total number of grid points and hence the operation count would be the same at each grid level. Such a scheme would negate one of the advantages of the multigrid method, namely, that all the computational work in performing relaxations on coarse grids was cheaper than comparable work on the finest grid. So he devised a special procedure which ensured that on coarse grids, the total number of grids points was reduced and less computational work was done. The resulting scheme was shown to be efficient in resolving some flows with alignment, but it appears to be rather complicated to implement, and it is doubtful that it will find its way into a general purpose computer code anytime soon.

Implementation of the multigrid method in time-stepping solution schemes for the compressible Navier-Stokes equations appear to be straightforward extension of that for Euler equations. Although, for the reasons given above, worse performance may be expected. Chima, Turkel and Schaffer (1987) compared implementations of three types of multigrid methods in explicit time-stepping multistage solution methods for Euler and Navier-Stokes equations. They found the Full multigrid-Full approximation storage (FMG-FAS) method proposed by Brandt (1977) to be the most efficient producing speedup factors of about 8.5 in the solution of the Euler equations for selected problems, but only about

2.1 in the solution of the Navier-Stokes equations. Multigrid schemes which use explicit time-stepping algorithm to solve the 3-D, compressible Navier-Stokes equations have also been reported by Arnone and Swanson (1988), Radespiel et. al (1990) and Swanson and Radespiel (1991). These are mostly central-differencing approximation methods, and the choice of artificial dissipation was found to be very important for efficient convergence. Yokota (1989) extended the previous implementation for the Euler equations (Yokota et. al, 1988) to the Reynolds-averaged, Navier-Stokes equations. The $k-\epsilon$ turbulence model was used to approximate the Reynolds stresses. Application to the calculation of the three-dimensional flow through blade passages showed convergence rates similar to those for the Euler equations. The use of wall-functions meant that the boundary layer need not be fully resolved so that grids with very high aspect ratios could be avoided, and hence, the lack of performance degradation. A novel method for solving the compressible, steady, Navier-Stokes equations was presented by Koren (1990). A first-order accurate upwind method with good smoothing properties was used for the discretization of the equations. Second-order accuracy was achieved through defect correction. The whole multigrid scheme exhibited good convergence characteristics in smooth flows, but somewhat poorer performance in non-smooth flows with shocks.

In the computation of flows in very complex geometries such as around multi-element airfoils or in complex inlet sections two approaches are popular; unstructured grids or multiple blocks of structured grids. Multigrid acceleration has also been achieved in solutions of the Euler and Navier-Stokes equations with either approach. Mavripilis (1988,1990) has demonstrated good multigrid convergence for the solution of the Euler equations on unstructured triangular meshes. Mavripilis and Jameson (1990) presented a similar implementation for the Navier-Stokes equations. Multigrid, multiblock methods

were presented for the Euler equations by Yadlin and Caughey (1991) and for the Navier-Stokes equations by Baysal et. al. (1991).

II: Stability Analysis of Numerical schemes for solving the Euler and Navier-Stokes Equations

Introduction

Von-Neumann stability analysis methods have been used to investigate convergence characteristics of the Euler and Navier-Stokes Equations. For the Euler equations, the Steger and Warming (1981) and Van Leer (1982) (flux-vector split) upwind schemes, and the Beam and Warming (1978) (ADI) central scheme were considered. However, for the Navier-Stokes equations, only the latter scheme was considered. In each of these cases, convergence characteristics of 1-D, 2-D and 3-D flow were investigated. Computations were mostly based on uniform flow fields. However, in order to examine the influence of variable flow fields, a local mode analysis was carried out using flow fields computed with the Proteus code for the viscous flow around a circular cylinder and the the transonic diffuser flow of Sajben.

II.1 Steger and Warming Upwind Scheme for Euler Equations

A stability analysis is presented here for the Steger-Warming flux-vector split upwind scheme for solving the Euler equations. The formulation and the results are similar to those presented by Anderson et al. (1988) for the van Leer flux-vector splitting scheme (Anderson et al., 1984, Hirsch, 1988).

II.1.1 Governing Equations

3-D Euler equations in Cartesian coordinates can be written as:

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = 0. \tag{1}$$

where

$$
\begin{aligned}
Q &= [\rho, \rho u, \rho v, \rho w, \rho E_T]^T \\
E &= [\rho u, \rho u^2 + p, \rho uv, \rho uw, (\rho E_T + p)u]^T \\
F &= [\rho v, \rho uv, \rho v^2 + p, \rho vw, (\rho E_T + p)v]^T \\
G &= [\rho w, \rho wu, \rho wv, \rho w^2 + p, (\rho E_T + p)w]^T
\end{aligned}
\tag{2}
$$

Time differencing of (1) yields

$$
\frac{Q^{n+1} - Q^n}{\Delta t} + \frac{\partial E^{n+1}}{\partial x} + \frac{\partial F^{n+1}}{\partial y} + \frac{\partial G^{n+1}}{\partial z} = 0. \tag{3}
$$

or in delta form

$$
\frac{\Delta Q^n}{\Delta t} + \frac{\partial E^{n+1}}{\partial x} + \frac{\partial F^{n+1}}{\partial y} + \frac{\partial G^{n+1}}{\partial z} = 0 \tag{4}
$$

where n refers to the old time level and n+1 to the new time level.

II.1.2 Linearized Equations

E^{n+1}, F^{n+1} and G^{n+1} in the above equations are non-linear functions of the conserved variables and are, linearized as follows:

$$
\begin{aligned}
E^{n+1} &= E^n + \frac{\partial E^n}{\partial t}\Delta t + O(\Delta t^2) \\
&= E^n + \frac{\partial E^n}{\partial Q}\frac{\partial Q^n}{\partial t}\Delta t + O(\Delta t^2) \\
&= E^n + A\Delta Q^n + O(\Delta t^2)
\end{aligned}
\tag{5}
$$

where

$$
A = \frac{\partial E^n}{\partial Q} \tag{6}
$$

is the flux-Jacobian.

Similarly,

$$F^{n+1} = F^n + B\Delta Q^n$$
$$G^{n+1} = G^n + C\Delta Q^n \tag{7}$$

where

$$B = \frac{\partial F^n}{\partial Q}, \quad C = \frac{\partial G^n}{\partial Q} \tag{8}$$

are the corresponding fluxes.

Hence, equation (4) becomes

$$\frac{\Delta Q^n}{\Delta t} + \frac{\partial}{\partial x}(E^n + A\Delta Q^n) + \frac{\partial}{\partial y}(F^n + B\Delta Q^n) + \frac{\partial}{\partial z}(G^n + C\Delta Q^n) = 0 \tag{9}$$

that is

$$(\frac{\mathbf{I}}{\Delta t} + \frac{\partial A^n}{\partial x} + \frac{\partial B^n}{\partial y} + \frac{\partial C^n}{\partial z})\Delta Q^n = -(\frac{\partial E^n}{\partial x} + \frac{\partial F^n}{\partial y} + \frac{\partial G^n}{\partial z}) \tag{10}$$

or

$$[\mathbf{I} + \Delta t(\delta_x A^n + \delta_y B^n + \delta_z C^n)]\Delta Q^n = -\Delta t(\delta_x E^n + \delta_y F^n + \delta_z G^n) \tag{11}$$

where

$$\delta_x = \frac{\partial}{\partial x}, \quad \delta_y = \frac{\partial}{\partial y}, \quad \delta_z = \frac{\partial}{\partial z} \tag{12}$$

and I is the identity matrix.

II.1.3 Flux-Vector Splitting

Based on the Steger and Warming flux-vector splitting, the fluxes A, B, C, E, F, and G can be split in the following manner:

$$\begin{aligned} A &= A^+ + A^- \quad B = B^+ + B^- \quad C = C^+ + C^- \\ E &= E^+ + E^- \quad F = F^+ + F^- \quad G = G^+ + G^- \end{aligned} \tag{13}$$

where

$$\begin{aligned} A^+ &= X_A D_A^+ X_A^{-1}, \;\; A^- = X_A D_A^- X_A^{-1} \\ B^+ &= X_B D_B^+ X_B^{-1}, \;\; B^- = X_B D_B^- X_B^{-1} \\ C^+ &= X_C D_C^+ X_C^{-1}, \;\; C^- = X_C D_C^- X_C^{-1} \end{aligned} \tag{14}$$

D_A^+ and D_A^- in the above transformation are diagonal matrices whose elements are the positive and negative eigenvalues of A, respectively, and the columns of X_A are the eigenvectors of A.

Linearized forms of the fluxes E^+, E^-, F^+, F^-, G^+, G^- are obtained from:

$$E^+ = A^+Q, \quad E^- = A^-Q$$
$$F^+ = B^+Q, \quad F^- = B^-Q \tag{15}$$
$$G^+ = C^+Q, \quad G^- = C^-Q$$

Equations (14) give approximate values for A^+, A^-, B^+, etc. Exact values for A^+, A^-, for examples, are obtained as follows:

$$A^+ = \frac{\partial E^+}{\partial Q} \quad , \quad A^- = \frac{\partial E^-}{\partial Q} \tag{16}$$

However, the present numerical solution uses the approximate forms, which are less expensive to compute.

II.1.4 Approximate Factorization.

The following equation is obtained if equations (13) are substituted in (11):

$$\begin{aligned}[\mathbf{I} + \Delta t\delta x(A^+ + \mathrm{A}^-) + \Delta t\delta y(B^+ + B^-) + \Delta t\delta z(C^+ + C^-)]\Delta Q^n = \\ - \Delta t[\delta x(E^+ + E^-) + \delta y(F^+ + F^-) + \delta z(G^+ + G^-)]\end{aligned} \tag{17}$$

Assuming that δ^+ and δ^- stand for forward and backward differencing respectively, then, for proper information transfer, it is customary to write

$$\begin{aligned}[\mathbf{I} + \Delta t(\delta_x^- A^+ + \delta_x^+ A^-) + \Delta t(\delta_y^- B^+ + \delta_y^+ B^-) + \Delta t(\delta_z^- C^+ + \delta_z^+ C^-)]\Delta Q = \\ - \Delta t[\delta_x^- E^+ + \delta_x^+ E^- + \delta_y^- F^+ + \delta_y^+ F^- + \delta_z^- G^+ + \delta_z^+ G^-]\end{aligned} \tag{18}$$

Equation (18) can be factorized approximately into any of the following expressions:

$$(i)\quad [\mathbf{I} + \Delta t(\delta_x^- A^+ + \delta_x^+ A^-)][I + \Delta t(\delta_y^- B^+ + \delta_y^+ B^-)] \\ [I + \Delta t(\delta_z^- C^+ + \delta_z^+ C^-)]\Delta Q^n = -\Delta t R^n \tag{19}$$

$$(ii)\quad [\mathbf{I} + \Delta t(\delta_x^- A^+ + \delta_y^- B^+ + \delta_z^- C^+)] \\ [I + \Delta t(\delta_x^+ A^- + \delta_y^+ B^- + \delta_z^+ C^-)]\Delta Q^n = -\Delta t R^n \tag{20}$$

$$(iii)\quad [\mathbf{I} + \Delta t(\delta_x^- A^+ + \delta_x^+ A^- + \delta_z^- C^+)] \\ [I + \Delta t(\delta_y^- B^+ + \delta_y^- B^- + \delta_z^+ C^-)]\Delta Q^n = -\Delta t R^n \tag{21}$$

where

$$R^n = \delta_x^- E^+ + \delta_x^+ E^- + \delta_y^- F^+ + \delta_y^+ F^- + \delta_z^- G^+ + \delta_z^+ G^- \tag{22}$$

Following Anderson et al. (1988) the splits in (19), (20), and (21), respectively are called spatial split, eigenvalue split and combination split. The last two splits fall under the category of the LU decomposition.

Substituting equations (15) into (22), while holding the fluxes locally constant, we have:

$$\begin{aligned} R^n &= A^+\delta_x^- Q^n + A^-\delta_x^+ Q^n + B^+\delta_y^- Q^n + B^-\delta_y^+ Q^n + C^+\delta_z^- Q^n + C^-\delta_z^+ Q^n \\ &= [A^+\delta_x^- + A^-\delta_x^+ + B^+\delta_y^- + B^-\delta_y^+ + C^+\delta_z^- + C^-\delta_z^+]Q^n \end{aligned} \tag{23}$$

II.1.5 Fourier symbols

Any of the equations (19), (20), and (21) can be written as

$$N\Delta Q^n = -L = \Delta t R^n \tag{24}$$

von Neumann stability analysis is used on this system of linear equations (18) by letting

$$Q^n = \lambda^n U_0 e^{Ii\phi_x} e^{Ij\phi_y} e^{Ik\phi_z} \tag{25}$$

where $I = \sqrt{-1}$, λ is the amplification factor, and ϕ_x, ϕ_y, ϕ_z represent the modes in the x,y,z direction, respectively.

The Fourier representation of equation (18) is obtained for each of the splits as follows:

(i) Spatial split

Using (25), $N\Delta Q^n$ of equation (24) can be factorized as follows:

$$\begin{aligned} N\Delta Q^n &= N(Q^{n+1} - Q^n) \\ &= \hat{N}(\lambda^{n+1} - \lambda^n) U_0 e^{Ii\phi_x} e^{Ij\phi_y} e^{Ik\phi_z} \\ &= \hat{N}(\lambda - 1)\lambda^n U_0 e^{Ii\phi_x} e^{Ij\phi_y} e^{Ik\phi_z} \\ &= \hat{N}(\lambda - 1) Q^n \end{aligned} \tag{26}$$

$\hat{N}$ is the Fourier symbol of N.

Thus

$$N = [\mathbf{I}+\Delta t(\delta_x^- A^+ + \delta_x^+ A^-)][\mathbf{I}+\Delta t(\delta_y^- B^+ + \delta_y^+ B^-)][\mathbf{I} + \Delta t(\delta_z^- C^+ + \delta_z^+ C^-)] \quad (27)$$

and

$$\begin{aligned}\hat{N} =&\{\mathbf{I} + \frac{\Delta t}{\Delta x}[A^+\left(1 - e^{-I\phi_x}\right) + A^-\left(e^{I\phi_x} - 1\right)]\}\\ &\{\mathbf{I} + \frac{\Delta t}{\Delta y}[B^+\left(1 - e^{-I\phi_y}\right) + B^-\left(e^{I\phi_y} - 1\right)]\}\\ &\{\mathbf{I} + \frac{\Delta t}{\Delta z}[C^+\left(1 - e^{-I\phi_z}\right) + C^-\left(e^{I\phi_z} - 1\right)]\}\end{aligned} \quad (28)$$

Here, we have used first order accuracy for this implicit term. However, the R^n expression is evaluated using second order accuracy as follows:

$$\begin{aligned}\delta^- u_i &= (3u_i - 4u_{i-1} + u_{i-2})/2\Delta x\\ \delta^+ u_i &= (-3u_i + 4u_{i+1} - u_{i+2})/2\Delta x\end{aligned} \quad (29)$$

Using these in (23) yields

$$\begin{aligned}R^n = \{&\frac{1}{2\Delta x}[A^+(3 - 4e^{-I\phi_x} + e^{-2I\phi_x})+A^-(-3 + 4e^{I\phi_x} - e^{2I\phi_x})]+\\ &\frac{1}{2\Delta y}[B^+(3 - 4e^{-I\phi_y} + e^{-2I\phi_y})+B^-(-3 + 4e^{I\phi_y} - e^{2I\phi_y})]+\\ &\frac{1}{2\Delta z}[C^+(3 - 4e^{-I\phi_z} + e^{-2I\phi_z})+C^-(-3 + 4e^{I\phi_z} - e^{2I\phi_z})]\}Q^n\end{aligned} \quad (30)$$

but from (24),

$$-L = \Delta t R^n \quad (31)$$

Setting $L = \hat{L}Q^n$ therefore implies

$$\begin{aligned}\hat{-L} = \{&\frac{\Delta t}{2\Delta x}[A^+(3 - 4e^{-I\phi_x} + e^{-2I\phi_x}) + A^-(-3 + 4e^{I\phi_x} - e^{2I\phi_x})]+ \\ &\frac{\Delta t}{2\Delta y}[B^+(3 - 4e^{-I\phi_y} + e^{-2I\phi_y}) + B^-(-3 + 4e^{I\phi_y} - e^{2I\phi_y})]+ \\ &\frac{\Delta t}{2\Delta z}[C^+(3 - 4e^{-I\phi_z} + e^{-2I\phi_z}) + C^-(-3 + 4e^{I\phi_z} - e^{2I\phi_z})]\}\end{aligned} \tag{32}$$

$\hat{L}$ is the Fourier symbol of L and since R^n is the same in the three splits, equation (32) also expresses the Fourier symbol $\hat{L}$ for the respective L in the eigenvalue split and the combination split.

Substituting equations (26) and (31) into equation (24) we have

$$\begin{aligned} N\Delta Q &= -L \\ \hat{N}(\lambda - 1)\mathbf{v} &= -\hat{L}\mathbf{v} \end{aligned} \tag{33}$$

or in general

$$\hat{N}(\lambda - 1)\mathbf{v} = -\hat{L}\mathbf{v} \tag{34}$$

$$\begin{aligned} \left(\hat{N} - \hat{L}\right)\mathbf{v} &= \lambda\hat{N}\mathbf{v} \\ i.e. \qquad K\mathbf{v} &= \lambda\hat{N}\mathbf{v} \qquad if \quad K = \hat{N} - \hat{L} \end{aligned} \tag{35}$$

This is a generalized eigenvalue problem that can be solved using the standard routines such as in the IMSL mathematical library.

Note that with

$$\begin{aligned} e^{I\phi} &= \cos\phi + I\sin\phi \\ e^{-I\phi} &= \cos\phi - I\sin\phi \end{aligned} \tag{36}$$

the Fourier symbols $\hat{N}$ *and* $\hat{L}$, can be written as

$$\begin{aligned} \hat{N} = &\left\{\mathbf{I} + \frac{\Delta t}{\Delta x}\left[(A^+ - A^-)(1-\cos\phi_x) + (A^+ + A^-)I\sin\phi_x\right]\right\} \\ &\left\{\mathbf{I} + \frac{\Delta t}{\Delta y}\left[(B^+ - B^-)(1-\cos\phi_y) + (B^+ + B^-)I\sin\phi_y\right]\right\} \\ &\left\{\mathbf{I} + \frac{\Delta t}{\Delta z}\left[(C^+ - C^-)(1-\cos\phi_z) + (C^+ + C^-)I\sin\phi_z\right]\right\} \end{aligned} \tag{37}$$

$$\begin{aligned} \hat{L} = &\frac{\Delta t}{2\Delta x}\left[(A^+ - A^-)(3+\cos 2\phi_x - 4\cos\phi_x) + (A^+ + A^-)(4\sin\phi_x - \sin 2\phi_x)I\right] \\ &+\frac{\Delta t}{2\Delta y}\left[(B^+ - B^-)(3+\cos 2\phi_y - 4\cos\phi_y) + (B^+ + B^-)(4\sin\phi_y - \sin 2\phi_y)I\right] \\ &+\frac{\Delta t}{2\Delta z}\left[(C^+ - C^-)(3+\cos 2\phi_z - 4\cos\phi_z) + (C^+ + C^-)(4\sin\phi_z - \sin 2\phi_z)I\right] \end{aligned} \tag{38}$$

(ii) Eigenvalue Split

Analysis is similar to that carried out for the spatial split but with N now defined as

$$N = \left[\mathbf{I} + \Delta t(\delta_x^- A^+ + \delta_y^- B^+ + \delta_z^- C^+)\right]\left[\mathbf{I} + \Delta t(\delta_x^+ A^- + \delta_y^+ B^- + \delta_z^+ C^-)\right] \tag{39}$$

The associated Fourier symbol $\hat{N}$ is thus:

$$\hat{N} = \left[\mathbf{I} + \frac{\Delta t}{\Delta x}A^{+}\left(1 - e^{-I\phi_x}\right) + \frac{\Delta t}{\Delta y}B^{+}\left(1 - e^{-I\phi_y}\right) + \frac{\Delta t}{\Delta z}C^{+}\left(1 - e^{-I\phi_z}\right)\right] \left[\mathbf{I} + \frac{\Delta t}{\Delta x}A^{-}\left(e^{I\phi_x} - 1\right) + \frac{\Delta t}{\Delta y}B^{-}\left(e^{I\phi_y} - 1\right) + \frac{\Delta t}{\Delta z}C^{-}\left(e^{I\phi_z} - 1\right)\right] \tag{40}$$

or using equation (36)

$$\begin{aligned}\hat{N} = [\mathbf{I} + \frac{\Delta t}{\Delta x}A^{+}(1 - \cos\phi_x + I\sin\phi_x) + \frac{\Delta t}{\Delta y}B^{+}(1 - \cos\phi_y + I\sin\phi_y) + \\ \frac{\Delta t}{\Delta z}C^{+}(1 - \cos\phi_z + I\sin\phi_z)][I + \frac{\Delta t}{\Delta x}A^{-}(\cos\phi_x + I\sin\phi_x - 1) + \\ \frac{\Delta t}{\Delta y}B^{-}(\cos\phi_y + I\sin\phi_y - 1) + \frac{\Delta t}{\Delta z}C^{-}(\cos\phi_z + I\sin\phi_z - 1)]\end{aligned} \tag{41}$$

As remarked earlier, equation (38) also expresses the Fourier symbol $\hat{L}$ for this split.

(iii) Combination split

Here N is split as follows:

$$N = \left[\mathbf{I} + \Delta t(\delta_x^{-}A^{+} + \delta_x^{+}A^{-} + \delta_z^{-}C^{+})\right]\left[\mathbf{I} + \Delta t(\delta_y^{-}B^{+} + \delta_y^{+}B^{-} + \delta_z^{+}C^{-})\right] \tag{42}$$

The associated $\hat{N}$ is

$$\hat{N} = \left[\mathbf{I} + \frac{\Delta t}{\Delta x}A^{+}\left(1 - e^{-I\phi_x}\right) + \frac{\Delta t}{\Delta x}A^{-}\left(e^{I\phi_x} - 1\right) + \frac{\Delta t}{\Delta z}C^{+}\left(1 - e^{-I\phi_z}\right)\right] \left[\mathbf{I} + \frac{\Delta t}{\Delta y}B^{+}\left(1 - e^{-I\phi_y}\right) + \frac{\Delta t}{\Delta y}B^{-}\left(e^{I\phi_y} - 1\right) + \frac{\Delta t}{\Delta z}C^{-}\left(e^{I\phi_z} - 1\right)\right] \tag{43}$$

or

$$\hat{N} = [\mathbf{I} + \frac{\Delta t}{\Delta x}A^{+}(1 - \cos\phi_x + I\sin\phi_x) + \frac{\Delta t}{\Delta x}A^{-}(\cos\phi_x + I\sin\phi_x - 1) + \frac{\Delta t}{\Delta z}C^{+}(1 - \cos\phi_z + I\sin\phi_z)][\mathbf{I} + \frac{\Delta t}{\Delta y}B^{+}(1 - \cos\phi_y + I\sin\phi_y) + \frac{\Delta t}{\Delta y}B^{-}(\cos\phi_y + I\sin\phi_y - 1) + \frac{\Delta t}{\Delta z}C^{-}(\cos\phi_z + I\sin\phi_z - 1)] \tag{44}$$

Equation (38) also expresses the Fourier symbol $\hat{L}$ for this scheme.

II.1.6 CFL Condition.

The CFL condition for Euler equations can be written as

$$\Delta t = \frac{CFL}{\left[\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + \frac{|w|}{\Delta z} + a\sqrt{\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} + \frac{1}{(\Delta z)^2}}\right]} \tag{45}$$

Hence, assuming equal grid spacing , (Δt / Δx) is obtained from

$$\frac{\Delta t}{\Delta x} = \frac{CFL}{(|u| + |v| + |w| + a\sqrt{3})} \tag{46}$$

II.1.7 2–D and 1–D Cases

Following a similar analysis as carried out above for 3-D, the corresponding Fourier symbols $\hat{N}$ and $\hat{L}$, and the CFL conditions for 2-D and 1-D Euler equations using Steger and Warming upwind scheme can be writtten as follows:

2- D Case

Only two splits are possible here, the spatial split and the eigenvalue split. For the spatial split, we have

$$\hat{N} = \left\{\mathbf{I} + \frac{\Delta t}{\Delta x}\left[(A^+ - A^-)(1 - \cos\phi_x) + (A^+ + A^-) I \sin\phi_x\right]\right\} \left\{\mathbf{I} + \frac{\Delta t}{\Delta y}\left[(B^+ - B^-)(1 - \cos\phi_y) + (B^+ + B^-) I \sin\phi_y\right]\right\} \tag{47}$$

and

$$\hat{L} = \frac{\Delta t}{2\Delta x}\left[(A^+ - A^-)(3 + \cos 2\phi_x - 4\cos\phi_x) + (A^+ + A^-)(4\sin\phi_x - \sin 2\phi_x) I\right] + \frac{\Delta t}{2\Delta y}\left[(B^+ - B^-)(3 + \cos 2\phi_y - 4\cos\phi_y) + (B^+ + B^-)(4\sin\phi_y - \sin 2\phi_y) I\right] \tag{48}$$

For the eigenvalue split,

$$\hat{N} = [\mathbf{I} + \frac{\Delta t}{\Delta x} A^+(1 - \cos\phi_x + I\sin\phi_x) + \frac{\Delta t}{\Delta y} B^+(1 - \cos\phi_y + I\sin\phi_y)] [\mathbf{I} + \frac{\Delta t}{\Delta x} A^-(\cos\phi_x + I\sin\phi_x - 1) + \frac{\Delta t}{\Delta y} B^-(\cos\phi_y + I\sin\phi_y - 1)] \tag{49}$$

and $\hat{L}$ is as expressed in equation (48)

The CFL condition for this 2-D case is

$$\Delta t = \frac{CFL}{\left[\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + a\sqrt{\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2}}\right]} \tag{50}$$

or setting $\Delta x = \Delta y$,

$$\frac{\Delta t}{\Delta x} = \frac{CFL}{(|u| + |v| + a\sqrt{2})} \tag{51}$$

1-D Case

Only the spatial split is possible here. Thus the Fourier symbols are

$$\hat{N} = \left\{\mathbf{I} + \frac{\Delta t}{\Delta x}\left[(A^+ - A^-)(1 - \cos\phi_x) + (A^+ + A^-)I\sin\phi_x\right]\right\} \tag{52}$$

$$\hat{L} = \frac{\Delta t}{2\Delta x}\left[(A^+ - A^-)(3 + \cos 2\phi_x - 4\cos\phi_x) + (A^+ + A^-)(4\sin\phi_x - \sin 2\phi_x)I\right] \tag{53}$$

The CFL condition for this case is

$$\Delta t = \frac{\Delta x CFL}{|u| + a} \tag{54}$$

that is

$$\frac{\Delta t}{\Delta x} = \frac{CFL}{|u| + a} \tag{55}$$

II.1.8 Computational Procedures

A pseudo-code that implements the above analysis is presented in *Appendix A*. The code evaluates, for a particular CFL number, the maximum eigenvalue over all the frequencies, the average eigenvalue, the l_2 norm of the eigenvalues, etc.

Two averages are computed, λ_{avg} and λ_{avgmax}. The former is based on all the eigenvalues while the latter is obtained from only the maximum eigenvalue at each frequency. In the numerical examples, 16 frequencies were considered in each of the ranges $0 \leq \phi_x, \phi_y, \phi_z \leq 2\pi$.

In a standard relaxation scheme, the quantity of interest for stability and convergence is the maximum eigenvalue over the whole range which determines the limiting condition for linear stability. However, in multigrid scheme, the role of the relaxation scheme is not to reduce the total error but to smoothen it out i.e. reduce the high frequency modes. These fall in the range $(\frac{\pi}{2}, \frac{3\pi}{2})$.

Thus the smoothing factor is calculated from the following relation:

$$\lambda_\mu = max(|\lambda|) \tag{56}$$

for the high frequency modes in the range

$$\frac{\pi}{2} \leq (\phi_x, \phi_y, \phi_z) \leq \frac{3\pi}{2} \tag{57}$$

Flow variables assumed include $M_\infty = 0.8$, zero yaw and angle of attack and $\gamma = 1.4$. Also equal grid spacing in all directions was assumed.

The expressions for the fluxes E, F and G, from which Jacobians A, B and C are obtained, are shown in *Appendix B*. The symbolic manipulation software Mathematica was used to generate the Fortran code for computing these flux-Jacobians and thier splittings. The respective Mathematica sequence of commands is presented in *Appendix C*.

II.1.9 Results and Discussions

The average eigenvalue (λ_{avg}), the smoothing factor and the maximum eigenvalue for each of the splits for the 3-D, 2-D and 1-D cases based on the above analysis of Steger and Warming are shown in Figs. II.1.1-II.1.3. Computations have also been carried out based on Van Leer flux-vector splitting and the corresponding results are also shown in Figs. II.1.4-II.1.6. The 3-D results of the latter agree very well with that of Anderson et al. (1988). However, it appears that the spatial split for the Steger and Warming method does not have good smoothing properties as compared with the Van Leer spatial split. Based on the linear analysis, there is also a smaller range of CFL numbers over which it is stable.

In each of the two methods of flux-vector splitting, both the 2-D and 1-D cases are uncoditionally stable. On the other hand, the 3-D case is only stable for certain range of CFL numbers.

II.2 Beam and Warming Central Scheme for Euler Equations.

The foregoing analyses have been based on Steger-Warming upwind differencing . The analysis of the solution scheme based on Beam and Warming central differencing is presented here.

II.2.1 Governing Equations

3-D Euler equations in Cartesian coordinates can be written as:

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = 0. \tag{58}$$

where

$$\begin{aligned} Q &= [\rho, \rho u, \rho v, \rho w, \rho E_T]^T \\ E &= [\rho u, \rho u^2 + p, \rho uv, \rho uw, (\rho E_T + p)u]^T \\ F &= [\rho v, \rho uv, \rho v^2 + p, \rho vw, (\rho E_T + p)v]^T \\ G &= [\rho w, \rho wu, \rho wv, \rho w^2 + p, (\rho E_T + p)w]^T \end{aligned} \tag{59}$$

Time differencing of (58) yields

$$\frac{Q^{n+1} - Q^n}{\Delta t} + \frac{\partial E^{n+1}}{\partial x} + \frac{\partial F^{n+1}}{\partial y} + \frac{\partial G^{n+1}}{\partial z} = 0. \tag{60}$$

or in delta form

$$\frac{\Delta Q^n}{\Delta t} + \frac{\partial E^{n+1}}{\partial x} + \frac{\partial F^{n+1}}{\partial y} + \frac{\partial G^{n+1}}{\partial z} = 0 \tag{61}$$

II.2.2 Linearized Equations

E^{n+1}, F^{n+1} and G^{n+1} in the above equations are non-linear functions of the conserved variables and are, linearized as follows:

$$\begin{aligned} E^{n+1} &= E^n + \frac{\partial E^n}{\partial t}\Delta t + O(\Delta t^2) \\ &= E^n + \frac{\partial E^n}{\partial Q}\frac{\partial Q^n}{\partial t}\Delta t + O(\Delta t^2) \\ &= E^n + A\Delta Q^n + O(\Delta t^2) \end{aligned} \tag{62}$$

where

$$A = \frac{\partial E^n}{\partial Q} \tag{63}$$

is the flux-Jacobian.

Similarly,

$$\begin{aligned} F^{n+1} &= F^n + B\Delta Q^n \\ G^{n+1} &= G^n + C\Delta Q^n \end{aligned} \tag{64}$$

where

$$B = \frac{\partial F^n}{\partial Q}, \quad C = \frac{\partial G^n}{\partial Q} \tag{65}$$

are the corresponding fluxes.

Hence, equation (61) becomes

$$\frac{\Delta Q^n}{\Delta t} + \frac{\partial}{\partial x}(E^n + A\Delta Q^n) + \frac{\partial}{\partial y}(F^n + B\Delta Q^n) + \frac{\partial}{\partial z}(G^n + C\Delta Q^n) = 0 \tag{66}$$

that is

$$(\frac{\mathbf{I}}{\Delta t} + \frac{\partial A^n}{\partial x} + \frac{\partial B^n}{\partial y} + \frac{\partial C^n}{\partial z})\Delta Q^n = -(\frac{\partial E^n}{\partial x} + \frac{\partial F^n}{\partial y} + \frac{\partial G^n}{\partial z}) \tag{67}$$

or

$$[\mathbf{I} + \Delta t(\delta_x A^n + \delta_y B^n + \delta_z C^n)]\Delta Q^n = -\Delta t(\delta_x E^n + \delta_y F^n + \delta_z G^n) \tag{68}$$

where

$$\delta_x = \frac{\partial}{\partial x}, \quad \delta_y = \frac{\partial}{\partial y}, \quad \delta_z = \frac{\partial}{\partial z} \tag{69}$$

and $\mathbf{I}$ is the identity matrix.

II.2.3 Artificial Dissipation

To damp the high frequency waves that often characterize the central differencing scheme, dissipation terms are usually added. In this analysis, the following second-order implicit and fourth-order explicit artificial viscosities are employed.

$$\begin{aligned} D_x^i &= -\varepsilon_i \Delta t \Delta x \delta_{xx} \\ D_x^e &= -\varepsilon_e \Delta t \Delta x^3 \delta_{xxxx} \end{aligned} \tag{70}$$

With similar terms for y and z directions, equation (68) becomes

$$[\mathbf{I} + \Delta t(\delta_x A^n + \delta_y B^n + \delta_z C^n) - \varepsilon_I \Delta t(\Delta x \delta_{xx} + \Delta y \delta_{yy} + \Delta z \delta_{zz})]\Delta Q^n$$
$$= -\Delta t(\delta_x E^n + \delta_y F^n + \delta_z G^n) - \varepsilon_e \Delta t(\Delta x^3 \delta_{xxxx} + \Delta y^3 \delta_{yyyy} + \Delta z^3 \delta_{zzzz})Q^n \tag{71}$$

II.2.4 Approximate Factorization

Equation (71) can be factorized into the following expression:

$$[\mathbf{I} + \Delta t(\delta_x A^n - \varepsilon_I \Delta x \delta_{xx})][\mathbf{I} + \Delta t(\delta_y B^n - \varepsilon_I \Delta y \delta_{yy})]$$
$$[\mathbf{I} + \Delta t(\delta_z C^n - \varepsilon_I \Delta z \delta_{zz})]\Delta Q^n = R^n \tag{72}$$

where the linearized form of R^n is as follows:

$$R^n = -\Delta t[(A\delta_x + B\delta_y + C\delta_z) + \varepsilon_e(\Delta x^3 \delta_{xxxx} + \Delta y^3 \delta_{yyyy} + \Delta z^3 \delta_{zzzz})]Q^n \tag{73}$$

here A, B, C are assumed locally constant.

II.2.5 Fourier Symbols

von Neumann analysis similar to that carried out in section II.1.5 will also yield

$$\left(\hat{N} - \hat{L}\right)\mathbf{v} = \lambda \hat{N}\mathbf{v} \tag{74}$$

where equation (72) is written as

$$N\Delta Q = -L \tag{75}$$

$\hat{N}$ *and* $\hat{L}$ in equation (74) are the Fourier symbols of N and L. Using central differencing, these can be written as:

$$\hat{N} = [\mathbf{I} + \frac{\Delta t}{\Delta x}(AI\sin\phi_x + 4\varepsilon_I \sin^2\frac{\phi_x}{2})][\mathbf{I} + \frac{\Delta t}{\Delta y}(BI\sin\phi_y + 4\varepsilon_I \sin^2\frac{\phi_y}{2})][\mathbf{I} + \frac{\Delta t}{\Delta z}(CI\sin\phi_z + 4\varepsilon_I \sin^2\frac{\phi_z}{2})] \tag{76}$$

$$\hat{L} = \frac{A\Delta t}{\Delta x} I\sin\phi_x + \frac{B\Delta t}{\Delta y} I\sin\phi_y + \frac{C\Delta t}{\Delta z} I\sin\phi_z + 16\varepsilon_e\left(\frac{\Delta t}{\Delta x}\sin^4\frac{\phi_x}{2} + \frac{\Delta t}{\Delta y}\sin^4\frac{\phi_y}{2} + \frac{\Delta t}{\Delta z}\sin^4\frac{\phi_z}{2}\right) \tag{77}$$

Using the following Fourier signatures:

$$\begin{aligned} \Delta x\delta_x &= I\sin\phi_x \\ \Delta x^2\delta_{xx} &= -4\sin^2\frac{\phi_x}{2} \\ \Delta x^4\delta_{xxxx} &= 16\sin^4\frac{\phi_x}{2} \qquad etc \end{aligned} \tag{78}$$

II.2.6 2-D and 1-D Cases

The corresponding Fourier symbols for 2-D and 1-D Euler equations using Beam and Warming ADI scheme derived in the above manner are written as follows:

2-D Case

$$\hat{N} = [\mathbf{I} + \frac{\Delta t}{\Delta x}(AI\sin\phi_x + 4\varepsilon_I \sin^2\frac{\phi_x}{2})][\mathbf{I} + \frac{\Delta t}{\Delta y}(BI\sin\phi_y + 4\varepsilon_I \sin^2\frac{\phi_y}{2})] \tag{79}$$

$$\hat{L} = \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{B\Delta t}{\Delta y} I \sin\phi_y + 16\varepsilon_e \left(\frac{\Delta t}{\Delta x} \sin^4 \frac{\phi_x}{2} + \frac{\Delta t}{\Delta y} \sin^4 \frac{\phi_y}{2} \right) \qquad (80)$$

1-D Case

$$\hat{N} = [\mathbf{I} + \frac{\Delta t}{\Delta x}(AI \sin\phi_x + 4\varepsilon_I \sin^2 \frac{\phi_x}{2})] \qquad (81)$$

$$\hat{L} = \frac{\Delta t}{\Delta x}(AI \sin\phi_x + 16\varepsilon_e \sin^4 \frac{\phi_x}{2}) \qquad (82)$$

II.2.7 CFL Condition

The CFL conditions used here are similar to those imposed on the Steger and Warming upwind scheme.

II.2.8 Computational Procedures

A pseudo-code that implements the above analysis is presented in *Appendix D*. The code evaluates, for a particular CFL number, the maximum eigenvalue over all the frequencies, the average eigenvalue, the l_2 norm of the eigenvalues, etc.

Two averages are computed, λ_{avg} and λ_{avgmax}. The former is based on all the eigenvalues while the latter is obtained from the maximum eigenvalue of each frequency. In these numerical examples, 16 frequencies were considered in each of the ranges $0 \leq \phi_x, \phi_y, \phi_z \leq 2\pi$.

The smoothing factor is calculated from:

$$\lambda_\mu = max(|\lambda|) \tag{83}$$

for the high frequency modes in the range

$$\frac{\pi}{2} \leq (\phi_x, \phi_y, \phi_z) \leq \frac{3\pi}{2} \tag{84}$$

Flow variables assumed include $M_\infty = 0.8$, zero yaw and angle of attack and $\gamma = 1.4$. Also equal grid spacing in all directions was assumed.

The expressions for the fluxes E, F and G, from which Jacobians A, B and C are obtained, are also given in *Appendix B*.

II.2.9 Results and Discussions

The results obtained for the stability analysis of the 3-D Euler equations are shown in Figs. II.2.1–II.2.2. It can be seen from this figure that the Beam and Warming (ADI) scheme is only conditionally stable for CFL numbers below approximately 15. However, at lower CFL numbers, addition of dissipation reduces the amplification factor and considerably lowers the smoothing factor, thus making the scheme suitable for multigrid implementation. Whereas the lowest amplification factor is obtained at a CFL number of about 10, the best smoothing factors are obtained at CFL numbers closer to unity. Therefore for multigrid application the code will have to be run at much lower CFL numbers to obtain the best convergence rates.

Corresponding results for 2-D are shown in Figs. II.2.3. In this case, the scheme is stable for all CFL numbers and also suitable for multigrid implementation at lower CFL numbers. The 2-D results are quite similar to the 3-D ones But the 1-D results in Fig. II.2.4 give a different picture as the CFL number is increased, convergence should actually improve. It is clear from these results that 1-D analysis cannot be used to reliably predict multidimensional code behaviour.

II.3 3-D Navier-Stokes Equations

The stability of the full Navier-Stokes equations in three dimensions is examined here. Beam and Warming ADI central difference scheme is employed for the solution algorithm in line with the approach of the Proteus code.

II.3.1 Governing Equations.

3-D Navier-Stokes equations can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = \frac{\partial E_v}{\partial x} + \frac{\partial F_v}{\partial y} + \frac{\partial G_v}{\partial z}. \tag{85}$$

where

$$\begin{aligned}
Q &= [\rho, \rho u, \rho v, \rho w, \rho E_T]^T \\
E &= [\rho u, \rho u^2 + p, \rho uv, \rho uw, (\rho E_T + p)u]^T \\
F &= [\rho v, \rho uv, \rho v^2 + p, \rho vw, (\rho E_T + p)v]^T \\
G &= [\rho w, \rho wu, \rho wv, \rho w^2 + p, (\rho E_T + p)w]^T \\
E_v &= [0,\ \tau_{xx},\ \tau_{xy},\ \tau_{xz},\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - q_x]^T \\
F_v &= [0,\ \tau_{yx},\ \tau_{yy},\ \tau_{yz},\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - q_y]^T \\
G_v &= [0,\ \tau_{zx},\ \tau_{zy},\ \tau_{zz},\ u\tau_{zx} + v\tau_{zy} + w\tau_{zz} - q_z]^T
\end{aligned} \tag{86}$$

with τ_{xx}, τ_{yy} etc defined as follows:

$$
\begin{aligned}
\tau_{xx} &= 2\mu u_x + \lambda(u_x + v_y + w_z) \\
\tau_{yy} &= 2\mu v_y + \lambda(u_x + v_y + w_z) \\
\tau_{zz} &= 2\mu w_z + \lambda(u_x + v_y + w_z) \\
\tau_{xy} &= \mu(u_y + v_x) = \tau_{yx} \\
\tau_{yz} &= \mu(v_z + w_y) = \tau_{zy} \\
\tau_{zx} &= \mu(w_x + u_z) = \tau_{xz} \\
q_x &= -kT_x \\
q_y &= -kT_y \\
q_z &= -kT_z
\end{aligned}
\tag{87}
$$

The cross-derivatives that arise from these equations are difficult to handle numerically. For simplicity, therefore, we re-write equation (85), following the approach of Beam and Warming (1978) for 2-D, as follows:

$$
\begin{aligned}
\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = {} & \frac{\partial U_1}{\partial x} + \frac{\partial U_2}{\partial x} + \frac{\partial U_3}{\partial x} + \\
& \frac{\partial V_1}{\partial y} + \frac{\partial V_2}{\partial y} + \frac{\partial V_3}{\partial y} + \\
& \frac{\partial W_1}{\partial z} + \frac{\partial W_2}{\partial z} + \frac{\partial W_2}{\partial z}
\end{aligned}
\tag{88}
$$

where we have set

$$
\begin{aligned}
E_v(Q) &= U_1(Q, Q_x) + U_2(Q, Q_y) + U_3(Q, Q_z) \\
F_v(Q) &= V_1(Q, Q_x) + V_2(Q, Q_y) + V_3(Q, Q_z) \\
G_v(Q) &= W_1(Q, Q_x) + W_2(Q, Q_y) + W_3(Q, Q_z)
\end{aligned}
\tag{89}
$$

thus

$$U_1 + U_2 + U_3 = \begin{bmatrix} 0 \\ 2\mu u_x + \lambda(u_x + v_y + w_z) \\ \mu(u_y + v_x) \\ \mu(u_z + w_x) \\ \mu v(u_y + v_x) + \mu w(u_z + w_x) + \\ \lambda u(u_x + v_y + w_z) + 2\mu u u_x + kT_x \end{bmatrix} \tag{90}$$

$$V_1 + V_2 + V_3 = \begin{bmatrix} 0 \\ \mu(u_y + v_x) \\ 2\mu v_y + \lambda(u_x + v_y + w_z) \\ \mu(v_z + w_y) \\ \mu u(u_y + v_x) + \mu w(v_z + w_y) + \\ \lambda v(u_x + v_y + w_z) + 2\mu v v_y + kT_y \end{bmatrix} \tag{91}$$

$$W_1 + W_2 + W_3 = \begin{bmatrix} 0 \\ \mu(w_x + u_z) \\ \mu(v_z + w_y) \\ 2\mu w_z + \lambda(u_x + v_y + w_z) \\ \mu u(w_x + u_z) + \mu v(v_z + w_y) + \\ \lambda w(u_x + v_y + w_z) + 2\mu w w_z + kT_z \end{bmatrix} \tag{92}$$

Using Stokes hypothesis , $\lambda = -\frac{2}{3}\mu$ these reduce to

$$U_1 + U_2 + U_3 = \begin{bmatrix} 0 \\ \frac{2}{3}\mu(2u_x - v_y - w_z) \\ \mu(u_y + v_x) \\ \mu(u_z + w_x) \\ \mu v(u_y + v_x) + \mu w(u_z + w_x) + \\ \frac{2}{3}\mu u(2u_x - v_y - w_z) + kT_x \end{bmatrix} \tag{93}$$

$$V_1 + V_2 + V_3 = \begin{bmatrix} 0 \\ \mu(u_y + v_x) \\ \frac{2}{3}\mu(2v_y - u_x - w_z) \\ \mu(v_z + w_y) \\ \mu u(u_y + v_x) + \mu w(v_z + w_y) + \\ \frac{2}{3}\mu v(2v_y - u_x - w_z) + kT_y \end{bmatrix} \tag{94}$$

$$W_1 + W_2 + W_3 = \begin{bmatrix} 0 \\ \mu(w_x + u_z) \\ \mu(v_z + w_y) \\ \frac{2}{3}\mu(2w_z - v_y - u_x) \\ \mu u(w_x + u_z) + \mu v(v_z + w_y) + \\ \frac{2}{3}\mu w(2w_z - v_y - u_x) + kT_z \end{bmatrix} \tag{95}$$

Equation (89) further implies

$$U_1(Q, Q_x) = \begin{bmatrix} 0 \\ \frac{4}{3}\mu u_x \\ \mu v_x \\ \mu w_x \\ \mu v v_x + \mu w w_x + \frac{4}{3}\mu u u_x + kT_x \end{bmatrix} \tag{96}$$

$$U_2(Q, Q_y) = \begin{bmatrix} 0 \\ -\frac{2}{3}\mu v_y \\ \mu u_y \\ 0 \\ \mu v u_y - \frac{2}{3}\mu u v_y \end{bmatrix} \tag{97}$$

$$U_3(Q, Q_z) = \begin{bmatrix} 0 \\ -\frac{2}{3}\mu w_z \\ 0 \\ \mu u_z \\ \mu w u_z - \frac{2}{3}\mu u w_z \end{bmatrix} \tag{98}$$

$$V_1(Q, Q_x) = \begin{bmatrix} 0 \\ \mu v_x \\ -\frac{2}{3}\mu u_x \\ 0 \\ \mu u v_x - \frac{2}{3}\mu v u_x \end{bmatrix} \tag{99}$$

$$V_2(Q, Q_y) = \begin{bmatrix} 0 \\ \mu u_y \\ \frac{4}{3}\mu v_y \\ \mu w_y \\ \mu u u_y + \mu w w_y + \frac{4}{3}\mu v v_y + k T_y \end{bmatrix} \tag{100}$$

$$V_3(Q, Q_z) = \begin{bmatrix} 0 \\ 0 \\ -\frac{2}{3}\mu w_z \\ \mu v_z \\ \mu w v_z - \frac{2}{3}\mu v w_z \end{bmatrix} \tag{101}$$

$$W_1(Q, Q_x) = \begin{bmatrix} 0 \\ \mu w_x \\ 0 \\ -\frac{2}{3}\mu u_x \\ \mu u w_x - \frac{2}{3}\mu w u_x \end{bmatrix} \tag{102}$$

$$W_2(Q, Q_y) = \begin{bmatrix} 0 \\ 0 \\ \mu w_y \\ -\frac{2}{3}\mu v_y \\ \mu v w_y - \frac{2}{3}\mu w v_y \end{bmatrix} \tag{103}$$

$$W_3(Q, Q_z) = \begin{bmatrix} 0 \\ \mu u_z \\ \mu v_z \\ \frac{4}{3}\mu w_z \\ \mu u u_z + \frac{4}{3}\mu w w_z + \mu v v_z + k T_z \end{bmatrix} \tag{104}$$

In a turbulent flow μ will be the effective viscosity which is a sum of the molecular viscosity and the turbulent eddy viscosity given by a turbulence model. k will be the corresponding effective thermal diffusivity.

II.3.2 Beam and Warming ADI Scheme

The following generalized time-marching procedure of Beam-Warming (1978) is employed to advance equation (88)

$$\Delta Q^n = \frac{\theta_1}{1+\theta_2}\Delta t \frac{\partial(\Delta Q^n)}{\partial t} + \frac{\Delta t}{1+\theta_2}\frac{\partial Q^n}{\partial t} + \frac{\theta_2}{1+\theta_2}\Delta Q^{n-1} + O\left[\left(\theta_1 - \frac{1}{2} - \theta_2\right)\Delta t^2 + \Delta t^3\right] \tag{105}$$

From equation (88)

$$\frac{\partial(\Delta Q^n)}{\partial t} = -\left(\frac{\partial(\Delta E^n)}{\partial x} + \frac{\partial(\Delta F^n)}{\partial y} + \frac{\partial(\Delta G^n)}{\partial z}\right) + \frac{\partial(\Delta U_1^n)}{\partial x} + \frac{\partial(\Delta U_2^n)}{\partial x} + \frac{\partial(\Delta U_3^n)}{\partial x} + \frac{\partial(\Delta V_1^n)}{\partial y} + \frac{\partial(\Delta V_2^n)}{\partial y} + \frac{\partial(\Delta V_3^n)}{\partial y} + \frac{\partial(\Delta W_1^n)}{\partial z} + \frac{\partial(\Delta W_2^n)}{\partial z} + \frac{\partial(\Delta W_3^n)}{\partial z} \tag{106}$$

Substituting (88) and (105) into (106) yields:

$$\begin{aligned}\Delta Q^n = &\frac{\theta_1}{1+\theta_2}\Delta t[-\left(\frac{\partial(\Delta E^n)}{\partial x} + \frac{\partial(\Delta F^n)}{\partial y} + \frac{\partial(\Delta G^n)}{\partial z}\right) + \frac{\partial(\Delta U_1^n)}{\partial x} + \\ &\frac{\partial(\Delta U_2^n)}{\partial x} + \frac{\partial(\Delta U_3^n)}{\partial x} + \frac{\partial(\Delta V_1^n)}{\partial y} + \frac{\partial(\Delta V_2^n)}{\partial y} + \frac{\partial(\Delta V_3^n)}{\partial y} + \frac{\partial(\Delta W_1^n)}{\partial z} + \\ &\frac{\partial(\Delta W_2^n)}{\partial z} + \frac{\partial(\Delta W_3^n)}{\partial z}] + \frac{\Delta t}{1+\theta_2}[-\left(\frac{\partial E^n}{\partial x} + \frac{\partial F^n}{\partial y} + \frac{\partial G^n}{\partial z}\right) + \frac{\partial U_1^n}{\partial x} + \\ &\frac{\partial U_2^n}{\partial x} + \frac{\partial U_3^n}{\partial x} + \frac{\partial V_1^n}{\partial y} + \frac{\partial V_2^n}{\partial y} + \frac{\partial V_3^n}{\partial y} + \frac{\partial W_1^n}{\partial z} + \frac{\partial W_2^n}{\partial z} + \frac{\partial W_3^n}{\partial z}] + \\ &\frac{\theta_2}{1+\theta_2}\Delta Q^{n-1} + O\left[\left(\theta_1 - \frac{1}{2} - \theta_2\right)\Delta t^2 + \Delta t^3\right]\end{aligned} \tag{107}$$

i.e.

$$\begin{aligned}\Delta Q^n = &\frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial x}(-\Delta E^n + \Delta U_1^n + \Delta U_2^n + \Delta U_3^n) + \\ &\frac{\partial}{\partial y}(-\Delta F^n + \Delta V_1^n + \Delta V_2^n + \Delta V_3^n) + \frac{\partial}{\partial z}(-\Delta G^n + \Delta W_1^n + \Delta W_2^n + \Delta W_3^n)] \\ &+ \frac{\Delta t}{1+\theta_2}[\frac{\partial}{\partial x}(-E^n + U_1^n + U_2^n + U_3^n) + \frac{\partial}{\partial y}(-F^n + V_1^n + V_2^n + V_3^n) + \\ &\frac{\partial}{\partial z}(-G^n + W_1^n + W_2^n + W_3^n)] + \frac{\theta_2}{1+\theta_2}\Delta Q^{n-1} + \\ &O\left[\left(\theta_1 - \frac{1}{2} - \theta_2\right)\Delta t^2 + \Delta t^3\right]\end{aligned} \tag{108}$$

II.3.3 Linearized Equations

ΔE^n, ΔF^n and ΔG^n in the above formulations are linearized as follows:

$$E^{n+1} = E^n + \frac{\partial E^n}{\partial Q}\Delta Q^n + O(\Delta t^2) \tag{109}$$

i.e.

$$\Delta E^n \approx A\Delta Q^n \tag{110}$$

where $A = \frac{\partial E^n}{\partial Q}$ and$\Delta E^n = E^{n+1} - E^n$

Similarly for ΔF^n andΔG^n we have

$$\begin{aligned} \Delta F^n &\approx B\Delta Q^n, \quad B = \frac{\partial F^n}{\partial Q} \\ \Delta G^n &\approx C\Delta Q^n, \quad C = \frac{\partial G^n}{\partial Q} \end{aligned} \tag{111}$$

A, B and C are as defined in section II.2.

ΔU_1^n, ΔV_2^n and ΔW_3^n involve no cross derivative terms (that is $U_1 = U_1(Q, Q_x)$, $V_2 = V_2(Q, Q_y)$, and $W_3 = W_3(Q, Q_Z)$) and are therefore linearized as follows:

$$U_1^{n+1} = U_1^n + \frac{\partial U_1^n}{\partial Q}\Delta Q^n + \frac{\partial U_1^n}{\partial Q_x}\Delta Q_x + O(\Delta t^2) \tag{112}$$

Therefore,

$$\begin{aligned}\Delta U_1^n &\approx \frac{\partial U_1^n}{\partial Q}\Delta Q^n + \frac{\partial U_1^n}{\partial Q_x}\Delta Q_x \\ &= P^n \Delta Q + R^n \Delta Q_x \\ &= PQ^n + (R\Delta Q)_x^n - R_x \Delta Q^n \\ &= (P - R_x)\Delta Q^n + (R\Delta Q)_x^n \\ &= (P - R_x + \delta_x R)\Delta Q^n\end{aligned} \tag{113}$$

Here,

$$P = \frac{\partial U_1}{\partial Q}, \quad \underline{R} = \frac{\partial U_1}{\partial Q_x} \quad and \quad R_x = \frac{\partial R}{\partial x} \tag{114}$$

Similar analysis holds for ΔV_2^n and ΔW_3^n. Thus:

$$\begin{aligned}\Delta V_2^n &= (X - S_y)\Delta Q^n + (S\Delta Q)_y^n \\ &= (X - S_y + \delta_y S)\Delta Q^n\end{aligned} \tag{115}$$

and

$$\begin{aligned}\Delta W_3^n &= (K - Y_z)\Delta Q^n + (Y\Delta Q)_z^n \\ &= (K - Y_z + \delta_z Y)\Delta Q^n\end{aligned} \tag{116}$$

where

$$X = \frac{\partial V_2^n}{\partial Q}, \quad S = \frac{\partial V_2^n}{\partial Q_y}, \quad S_y = \frac{\partial S}{\partial y}$$
$$K = \frac{\partial W_3^n}{\partial Q}, \quad Y = \frac{\partial W_3^n}{\partial Q_z}, \quad Y_z = \frac{\partial Y}{\partial z} \tag{117}$$

ΔU_2^n, ΔU_3^n, ΔV_1^n, ΔV_3^n, ΔW_1^n and ΔW_2^n are the cross derivative terms and, in order to resolve the difficulties that will arise if they are linearized as above, they are evaluated explicitly. The following arguments ensure that this will lead to no loss of accuracy and that minimal computational effort is actually involved.

From Taylor series expansion,

$$\begin{aligned} U_2^{n+1} &= U_2^n + \frac{\partial U_2^n}{\partial t}\Delta t + \frac{\partial^2 U_2^n}{\partial t^2}(\Delta t)^2 + O(\Delta t)^3 \\ U_2^{n-1} &= U_2^n - \frac{\partial U_2^n}{\partial t}\Delta t + \frac{\partial^2 U_2^n}{\partial t^2}(\Delta t)^2 + O(\Delta t)^3 \\ \Rightarrow \quad U_2^{n+1} &+ U_2^{n-1} = 2U_2^n + O(\Delta t)^2 \end{aligned} \tag{118}$$

hence

$$U_2^{n+1} - U_2^n = U_2^n - U_2^{n-1} + O(\Delta t)^2 \tag{119}$$

or

$$\Delta U_2^n = \Delta U_2^{n-1} + O(\Delta t)^2 \tag{120}$$

Similarly,

$$\begin{aligned}
\Delta U_3^n &= \Delta U_3^{n-1} + O(\Delta t)^2 \\
\Delta V_1^n &= \Delta V_1^{n-1} + O(\Delta t)^2 \\
\Delta V_3^n &= \Delta V_3^{n-1} + O(\Delta t)^2 \\
\Delta W_1^n &= \Delta W_1^{n-1} + O(\Delta t)^2 \\
\Delta W_2^n &= \Delta W_2^{n-1} + O(\Delta t)^2
\end{aligned} \tag{121}$$

Substituting equations (110), (111), (113), (115), (116), (120) and (121) in (108) we have

$$\begin{aligned}
\Delta Q^n = &\frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial x}(-A+P-R_x)^n + \frac{\partial^2 R^n}{\partial x^2} + \frac{\partial}{\partial y}(-B+X-S_y)^n + \frac{\partial^2 S^n}{\partial y^2} + \\
&\frac{\partial}{\partial z}(-C+K-Y_z)^n + \frac{\partial^2 Y^n}{\partial z^2}]\Delta Q^n + \frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial x}(\Delta U_2^{n-1} + \Delta U_3^{n-1}) + \\
&\frac{\partial}{\partial y}(\Delta V_1^{n-1} + \Delta V_3^{n-1}) + \frac{\partial}{\partial z}(\Delta W_1^{n-1} + \Delta W_2^{n-1})] + \\
&\frac{\Delta t}{1+\theta_2}[\frac{\partial}{\partial x}(-E^n + U_1^n + U_2^n + U_3^n) + \frac{\partial}{\partial y}(-F^n + V_1^n + V_2^n + V_3^n) + \\
&\frac{\partial}{\partial z}(-G^n + W_1^n + W_2^n + W_3^n)] + \frac{\theta_2}{1+\theta_2}\Delta Q^{n-1} + \\
&O\left[\left(\theta_1 - \frac{1}{2} - \theta_2\right)\Delta t^2 + \Delta t^3\right]
\end{aligned} \tag{122}$$

Hence, the governing equation in linear form is as follows;

$$
\begin{aligned}
&\{\mathbf{I}+\frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial x}(A-P+R_x)^n-\frac{\partial^2 R^n}{\partial x^2}+\frac{\partial}{\partial y}(B-X+S_y)^n- \\
&\frac{\partial^2 S^n}{\partial y^2}+\frac{\partial}{\partial z}(C-K+Y_z)^n-\frac{\partial^2 Y^n}{\partial z^2}]\}\Delta Q^n \\
=&\frac{\overline{\theta}_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial x}(\Delta U_2^{n-1}+\Delta U_3^{n-1})+\frac{\partial}{\partial y}(\Delta V_1^{n-1}+\Delta V_3^{n-1})+ \\
&\frac{\partial}{\partial z}(\Delta W_1^{n-1}+\Delta W_2^{n-1})]+\frac{\Delta t}{1+\theta_2}[\frac{\partial}{\partial x}(-E^n+U_1^n+U_2^n+U_3^n)+ \\
&\frac{\partial}{\partial y}(-F^n+V_1^n+V_2^n+V_3^n)+\frac{\partial}{\partial z}(-G^n+W_1^n+W_2^n+W_3^n)]+ \\
&\frac{\theta_2}{1+\theta_2}\Delta Q^{n-1}+O\left[\left(\theta_1-\frac{1}{2}-\theta_2\right)\Delta t^2+\Delta t^3\right]
\end{aligned}
\tag{123}
$$

$\overline{\theta}_1$ is introduced for notation convenience and it is often set equal to zero for first order accurate schemes.

II.3.4 Approximate Factorization

Approximate factorization of (123) yields

$$
\begin{aligned}
&\{\mathbf{I}+\frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial x}(A-P+R_x)^n-\frac{\partial^2 R^n}{\partial x^2}]\} \\
&\{\mathbf{I}+\frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial y}(B-X+S_y)^n-\frac{\partial^2 S^n}{\partial y^2}]\} \\
&\{\mathbf{I}+\frac{\theta_1}{1+\theta_2}\Delta t[\frac{\partial}{\partial z}(C-K+Y_z)^n-\frac{\partial^2 Y^n}{\partial z^2}]\} \quad = RHS \quad (123)
\end{aligned}
\tag{124}
$$

Assuming constant flow properties, it can be shown that $(-P+R_x) = (-X+S_y) = (-K+Y_z) = 0$. Hence, using first order accurate, Euler-implicit scheme ($\overline{\theta}_1 = 0$, $\theta_2 = 0$, $\theta_1 = 1$) for equation (124) will yield:

$$[\mathbf{I} + \Delta t(\delta_x A - \delta_{xx} R)][\mathbf{I} + \Delta t(\delta_y B - \delta_{yy} S)][\mathbf{I} + \Delta t(\delta_z C - \delta_{zz} Y)]\Delta Q^n =$$
$$\Delta t[\delta_x(-E + U_1 + U_2 + U_3)^n + \delta_y(-F + V_1 + V_2 + V_3)^n + \qquad (125)$$
$$\delta_z(-G + W_1 + W_2 + W_3)] + O(\Delta t)^2$$

Assuming also that fluxes A, B, C etc., are locally constant, we can express E, U_1, U_2, U_3, F, V_1, etc, in the following linear form :

$$\begin{aligned} &E = AQ, \quad U_1 = \delta_x(RQ), \quad U_2 = \delta_y(R_1 Q), \quad U_3 = \delta_z(R_2 Q) \\ &F = BQ, \quad V_1 = \delta_x(S_1 Q), \quad V_2 = \delta_y(SQ), \quad V_3 = \delta_z(S_2 Q) \\ &G = CQ, \quad W_1 = \delta_x(Y_1 Q), \quad W_2 = \delta_y(Y_2 Q), \quad W_3 = \delta_z(YQ) \end{aligned} \qquad (126)$$

where

$$\begin{aligned} R &= \frac{\partial U_1}{\partial Q_x}, & R_1 &= \frac{\partial U_2}{\partial Q_y}, & R_2 &= \frac{\partial U_3}{\partial Q_z} \\ S_1 &= \frac{\partial V_1}{\partial Q_x}, & S &= \frac{\partial V_2}{\partial Q_y}, & S_2 &= \frac{\partial V_3}{\partial Q_z} \\ Y_1 &= \frac{\partial W_1}{\partial Q_x}, & Y_2 &= \frac{\partial W_2}{\partial Q_y}, & Y &= \frac{\partial W_3}{\partial Q_z} \end{aligned} \qquad (127)$$

Using these in equation (125) we have:

$$[\mathbf{I} + \Delta t(\delta_x A - \delta_{xx} R)][\mathbf{I} + \Delta t(\delta_y B - \delta_{yy} S)][\mathbf{I} + \Delta t(\delta_z C - \delta_{zz} Y)]\Delta Q^n =$$
$$-\Delta t[A\delta_x - R\delta_{xx} - R_1\delta_{yx} - R_2\delta_{zx} + B\delta_y - S_1\delta_{xy} - S\delta_{yy} - S_2\delta_{zy} + \qquad (128)$$
$$C\delta_z - Y_1\delta_{xz} - Y_2\delta_{yz} - Y\delta_{zz}]Q$$

II.3.5 Artificial Dissipation

To damp the high frequency waves that often characterize the central differencing scheme, dissipation terms are usually added. In this analysis, the following second-order implicit and fourth-order explicit artificial viscosities are employed.

$$\begin{aligned} D_x^i &= -\varepsilon_i \Delta t \Delta x \delta_{xx} \\ D_x^e &= -\varepsilon_e \Delta t \Delta x^3 \delta_{xxxx} \end{aligned} \tag{129}$$

With similar terms for y and z directions, equation (128) becomes

$$\begin{aligned} &[\mathbf{I} + \Delta t(\delta_x A - \delta_{xx} R) - \varepsilon_I \Delta t \Delta x \delta_{xx}][\mathbf{I} + \Delta t(\delta_y B - \delta_{yy} S) - \varepsilon_I \Delta t \Delta y \delta_{yy}] \\ &\qquad [\mathbf{I} + \Delta t(\delta_z C - \delta_{zz} Y) - \varepsilon_I \Delta t \Delta z \delta_{zz}] \Delta Q^n = \\ &-\Delta t[A\delta_x - R\delta_{xx} - R_1\delta_{yx} - R_2\delta_{zx} + B\delta_y - S_1\delta_{xy} - S\delta_{yy} - S_2\delta_{zy} + C\delta_z \\ &- Y_1\delta_{xz} - Y_2\delta_{yz} - Y\delta_{zz} + \varepsilon_e \Delta x^3 \delta_{xxxx} + \varepsilon_e \Delta y^3 \delta_{yyyy} + \varepsilon_e \Delta z^3 \delta_{zzzz}] Q^n \end{aligned} \tag{130}$$

II.3.6 Fourier Symbols

von Neumann analysis similar to that carried out in section II.1.5 will also yield

$$\left(\hat{N} - \hat{L}\right)\mathbf{v} = \lambda \hat{N}\mathbf{v} \tag{131}$$

where equation (130) is written as

$$N\Delta Q = -L \tag{132}$$

$\hat{N}$ *and* $\hat{L}$ in equation (131) are the Fourier symbols of N and L. Using central differencing, these can be written as:

$$\hat{N} = \left[\mathbf{I} + \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{4R\Delta t}{\Delta x^2}\sin^2\frac{\phi_x}{2} + 4\frac{\Delta t}{\Delta x}\varepsilon_I \sin^2\frac{\phi_x}{2}\right]$$
$$\left[\mathbf{I} + \frac{B\Delta t}{\Delta y} I \sin\phi_y + \frac{4S\Delta t}{\Delta y^2}\sin^2\frac{\phi_y}{2} + 4\frac{\Delta t}{\Delta y}\varepsilon_I \sin^2\frac{\phi_y}{2}\right] \tag{133}$$
$$\left[\mathbf{I} + \frac{C\Delta t}{\Delta z} I \sin\phi_z + \frac{4Y\Delta t}{\Delta z^2}\sin^2\frac{\phi_z}{2} + 4\frac{\Delta t}{\Delta z}\varepsilon_I \sin^2\frac{\phi_z}{2}\right]$$

$$\begin{aligned}\hat{L} =& \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{4R\Delta t}{\Delta x^2}\sin^2\frac{\phi_x}{2} + \frac{R_1\Delta t}{\Delta x\Delta y}\sin\phi_x \sin\phi_y + \\ & \frac{R_2\Delta t}{\Delta x\Delta z}\sin\phi_x \sin\phi_z + \frac{B\Delta t}{\Delta y} I \sin\phi_y + \frac{S_1\Delta t}{\Delta x\Delta y}\sin\phi_x \sin\phi_y \\ & + \frac{4S\Delta t}{\Delta y^2}\sin^2\frac{\phi_y}{2} + \frac{S_2\Delta t}{\Delta y\Delta z}\sin\phi_y \sin\phi_z + \frac{C\Delta t}{\Delta z} I \sin\phi_z + \\ & \frac{Y_1\Delta t}{\Delta x\Delta z}\sin\phi_x \sin\phi_z + \frac{Y_2\Delta t}{\Delta y\Delta z}\sin\phi_y \sin\phi_z + \frac{4Y\Delta t}{\Delta z^2}\sin^2\frac{\phi_z}{2} \\ & + 16\varepsilon_e\left(\frac{\Delta t}{\Delta x}\sin^4\frac{\phi_x}{2} + \frac{\Delta t}{\Delta y}\sin^4\frac{\phi_y}{2} + \frac{\Delta t}{\Delta z}\sin^4\frac{\phi_z}{2}\right)\end{aligned} \tag{134}$$

II.3.7 2-D and 1-D Cases

The corresponding Fourier symbols for 2-D and 1-D Navier-Stokes equations using Beam and Warming ADI scheme derived in the above manner are written as follows:

2-D Case

$$\hat{N} = \left[\mathbf{I} + \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{4R\Delta t}{\Delta x^2}\sin^2\frac{\phi_x}{2} + 4\frac{\Delta t}{\Delta x}\varepsilon_I \sin^2\frac{\phi_x}{2}\right]$$
$$\left[\mathbf{I} + \frac{B\Delta t}{\Delta y} I \sin\phi_y + \frac{4S\Delta t}{\Delta y^2}\sin^2\frac{\phi_y}{2} + 4\frac{\Delta t}{\Delta y}\varepsilon_I \sin^2\frac{\phi_y}{2}\right] \tag{135}$$

$$\hat{L} = \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{4R\Delta t}{\Delta x^2} \sin^2\frac{\phi_x}{2} + \frac{R_1\Delta t}{\Delta x \Delta y} \sin\phi_x \sin\phi_y + \frac{B\Delta t}{\Delta y} I \sin\phi_y + \frac{S_1\Delta t}{\Delta x \Delta y} \sin\phi_x \sin\phi_y + \frac{4S\Delta t}{\Delta y^2} \sin^2\frac{\phi_y}{2} + 16\varepsilon_e \left(\frac{\Delta t}{\Delta x} \sin^4\frac{\phi_x}{2} + \frac{\Delta t}{\Delta y} \sin^4\frac{\phi_y}{2} \right) \tag{136}$$

1-D Case

$$\hat{N} = \left[\mathbf{I} + \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{4R\Delta t}{\Delta x^2} \sin^2\frac{\phi_x}{2} + 4\frac{\Delta t}{\Delta x}\varepsilon_I \sin^2\frac{\phi_x}{2} \right] \tag{137}$$

$$\hat{L} = \frac{A\Delta t}{\Delta x} I \sin\phi_x + \frac{4R\Delta t}{\Delta x^2} \sin^2\frac{\phi_x}{2} + 16\frac{\Delta t}{\Delta x}\varepsilon_e \sin^4\frac{\phi_x}{2} \tag{138}$$

II.3.7 Reynolds Number and CFL Condition

The CFL conditions used here are similar to those imposed on the Euler schemes and the Reynolds number for the 3-D case is defined as follows:

$$Re = \frac{\rho(\sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2})(\sqrt{u^2 + v^2 + w^2})}{\mu_l} \tag{139}$$

II.3.8 Computational Procedures

A pseudo-code that implements the above analysis is presented in *Appendix E*. The code evaluates, for a particular CFL number, the maximum eigenvalue over all the frequencies, the average eigenvalue, the l_2 norm of the eigenvalues, etc.

Two averages are computed, λ_{avg} and λ_{avgmax}. The former is based on all the eigenvalues while the latter is obtained from the maximum eigenvalue of each frequency. In these numerical examples, 16 frequencies were considered in each of the ranges $0 \leq \phi_x, \phi_y, \phi_z \leq 2\pi$.

The smoothing factor is calculated from:

$$\lambda_\mu = max(|\lambda|) \tag{140}$$

for the high frequency modes in the range

$$\frac{\pi}{2} \leq (\phi_x, \phi_y, \phi_z) \leq \frac{3\pi}{2} \tag{141}$$

Flow variables assumed include $M_\infty = 0.8$, zero yaw and angle of attack and γ = 1.4. Also equal grid spacing in all directions was assumed.

The fluxes E, F and G, and hence, the Jacobians A, B and C are the same as for the Euler equation already given in section II.2. On the other hand, expressions for U_1, U_2, U_3, V_1, V_2, V_3, W_1, W_2, W_3 from which the Jacobians R, R_1, R_2, S, S_1, S_2, Y, Y_1, Y_2 are obtained are shown in *Appendix F*. The Math-

ematica sequence of commands that are used in generating the Jacobians is presented in *Appendix G.*

II.3.9 Results and Discussions

Results for the stability analysis for the 3-D Navier-Stokes equations are shown in Figs. II.3.1 - II.3.8. The stability was examined at Reynolds numbers of 100 and 1000000, and at different levels of dissipation. It could be observed from the figures that for a given level of disspation, the scheme is more stable at lower Reynolds numbers. Generally, addition of dissipation reduces the amplification factor and the smoothing factor at lower CFL numbers.

Results for the influence of Reynolds number and dissipation on the stability of the 2-D Navier-Stokes equations are shown in Figs. II.3.8 - II.3.12. Also for the given level of dissipation $\varepsilon_i = 1.4$, $\varepsilon_e = 2.8$, the effect of Mach number was examined and the results are presented in Figs. II.3.13 - II.3.16. The scheme is less stable at lower Mach numbers, but the smoothing factor is little affected. These indicate that convergence rates in single grid schemes are expected to deteriorate rapidly as the Mach number is lowered towards incompressible flow regimes, but with multigrid the overall convergence rates should not be much worse. Corresponding results showing the effect of Reynolds number and dissipation on the stability of Beam and Warming (ADI) scheme for 1-D Navier-Stokes equations are shown in Figs. II.3.17 - II.3.19. In addition, the results of the amplification factor over the frequency range $0 \leq \phi_x, \phi_y \leq \pi$ for CFL = 2.5 are presented in Figs. II.3.20 - II.3.25. Without artificial dissipation high frequencing smoothing is quite poor, but with the usual form of constant dissipation terms good high frequency damping is achieved.

II.4 Local Mode Analysis

The analysis carried out in previous sub-sections are based on uniform flow fields. This gives an indication of general stability characteristics of the numerical schemes, but insufficient information as to the performance in practical problems. In a multigrid scheme the role of relaxation is not to reduce the error but to smoothen it out, i.e., reduce the high frequency modes. This is essentially a local process since the high frequency modes have short wavelengths and are thus spatially decoupled. The lower frequency modes will of course be reduced on coarser grids. Hence, following Brandt (1977) we can carry out a local mode analysis based on frozen coefficients and fluxes obtained from actual computations to obtain reliable predictions of multigrid performance in practical flow situations. Such a local mode analysis of two practical problems based on the actual flow fields are examined here. These problems are the viscous flow over a circular cylinder and the Sajben transonic diffuser flow. The former was computed in the Proteus code at a Reynolds number of 20 and CFL number of 10 while the latter was run at a Reynolds number of about 120000 and CFL number of 5. Constant dissipation ($\varepsilon_i = 2, \quad \varepsilon_e = 1$) was assumed in the viscous flow while the non-linear Jameson type dissipation was used in the Sajben case. Stability and convergence analyses based on actual flow fields obtained from the Proteus code were performed for each of these cases.

II.4.1 Jameson Non-Linear Dissipation

The implementation of constant dissipation coefficients in the stability analysis was presented in section II.2.3. However, since the Sajben transonic diffuser test case was run with the non-linear dissipation formulation, we follow here a similar approach. The dissipative terms are strictly explicit and are combinations of second- and fourth-

differences. As stated by Caughey (1988), the second-difference terms dissipate spurious waves in the shock region and fourth-difference terms are employed for steady state convergence.

In operator form, the dissipation in x-direction is as follows:

$$\begin{aligned} D_x^e &= \Delta x \delta_x \left[\psi_x \left(\varepsilon_x^2 \Delta x \delta_x - \varepsilon_x^4 \Delta x^3 \delta_{xxx} \right) \right] \\ &= \psi_x \left(\varepsilon_x^2 \Delta x^2 \delta_{xx} - \varepsilon_x^4 \Delta x^4 \delta_{xxxx} \right) \\ &= -\psi_x \left(4\varepsilon_x^2 \sin^2 \frac{\phi_x}{2} + 16\varepsilon_x^4 \sin^4 \frac{\phi_x}{2} \right) \end{aligned} \tag{142}$$

Similarly for y-direction,

$$\begin{aligned} D_y^e &= \Delta y \delta_y \left[\psi_y \left(\varepsilon_y^2 \Delta y \delta_y - \varepsilon_y^4 \Delta y^3 \delta_{yyy} \right) \right] \\ &= -\psi_y \left(4\varepsilon_y^2 \sin^2 \frac{\phi_y}{2} + 16\varepsilon_y^4 \sin^4 \frac{\phi_y}{2} \right) \end{aligned} \tag{143}$$

The parameters ε^2 and ε^4 are the coefficients for the second- and fourth-difference artificial dissipation. Rather than being constants as in the previous analysis, they are computed as functions of the pressure field as follows.

$$\begin{aligned} \left(\varepsilon_x^2\right)_i &= \kappa_2 \Delta t \; max\left(\sigma_{i+1,j}^x \, , \; \sigma_{i,j}^x \, , \; \sigma_{i-1,j}^x\right) \\ \left(\varepsilon_x^4\right)_i &= max\left[0, \; \kappa_4 \Delta t - \left(\varepsilon_x^2\right)_i\right] \\ \left(\varepsilon_y^2\right)_j &= \kappa_2 \Delta t \; max\left(\sigma_{i,j+1}^y \, , \; \sigma_{i,j}^y \, , \; \sigma_{i,j-1}^y\right) \\ \left(\varepsilon_y^4\right)_j &= max\left[0, \; \kappa_4 \Delta t - \left(\varepsilon_y^2\right)_j\right] \end{aligned} \tag{144}$$

where

$$\sigma^x_{i,j} = \left| \frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{p_{i+1,j} + 2p_{i,j} + p_{i-1,j}} \right|$$
$$\sigma^y_{i,j} = \left| \frac{p_{i,j+1} - 2p_{i,j} + p_{i,j-1}}{p_{i,j+1} + 2p_{i,j} + p_{i,j-1}} \right| \tag{145}$$

ψ_x and ψ_y, the spectra radii, are assumed locally constant and they are expressed as follows:

$$\psi_x = \left(\frac{|u| + a}{\Delta x} + \frac{|v| + a}{\Delta y} \right)_{i+1,j} + \left(\frac{|u| + a}{\Delta x} + \frac{|v| + a}{\Delta y} \right)_{i,j}$$
$$\psi_y = \left(\frac{|u| + a}{\Delta x} + \frac{|v| + a}{\Delta y} \right)_{i,j+1} + \left(\frac{|u| + a}{\Delta x} + \frac{|v| + a}{\Delta y} \right)_{i,j} \tag{146}$$

a is the speed of sound, κ_2 and κ_4 are problem-dependent constants and for the Sajben case κ_2 = 0.1 and κ_4 = 0.0004.

II.4.2 Results and Discussions

Contours of the computed amplification factors for the viscous flow past a cylinder are shown in Fig. II.4.1, and for the smoothing factors in Fig. II.4.2. The convergence rate for this problem was acceptable in practice, and the predicted amplification factors are actually similar to the computed residual growth rate presented in Appendix H. The smoothing factors are lower than the amplification factors so multigrid should improve the overall rate of convergence.

Contours of the amplification factors and smoothing rates obtained from the local mode analysis of the Sajben transonic flow case are presented in Figs. II.4.3 and II.4.4, respectively. In this case the amplification and smoothing factors are both very close to

unity, which explains the poor asymptotic convergence. Also multigrid is not expected to produce better convergence. The reason for this is likely to be the use of strictly explicit dissipation. Good smoothing factors can be obtained by introducing implicit dissipation as shown in Figs. II.4.5 and II.4.6 which were run with constant implicit and explicit disspation. From the results shown in Fig. II.3.2, it would be expected that good smoothing factors can also be achieved by using lower CFL numbers of about 1 to 2. Amplification and smoothing factors computed with a CFL number of 1 are presented in Figs. II.4.7-II.4.10. These show that contrary to expectations the smoothing factors are worse than those obtained with a CFL number of 5. The explanation for this is shown in Fig. II.4.11 where the stability analysis results for different grid cell aspect ratios are presented. As the aspect ratio is increased to 10, the CFL number for optimum smoothing increases from about 1 to 4. This emphasizes the role of aspect ratio in the stability analysis. For the Sajben case, the grid cell aspect ratio varies between 4 and 10 (see Fig. III.25). These results underscore the importance of local mode analysis based on actual frozen coefficients in predicting the convergence characteristics of numerical schemes.

III : Convergence Analysis of Proteus Test Problems.

Introduction

Convergence analysis was carried out for the 2-D steady flow problems presented in the Proteus User's Guide to determine the sensitivity to grid refinement. These are flow past a circular cylinder, turbulent flow over a flat plate and Sajben transonic diffuser flow. The first is axi-symmetric and the others are plane flows. Both inviscid and viscous (laminar) flow over a circular cylinder were computed. The other two test cases were turbulent flow. The second test case used both an algebraic turbulence model and a two-equation model but only the algebraic model was used in the third test case.

Basic iterative methods, including time-stepping approaches, have convergence rates which display a sensitivity to grid size of the form $1 - O(h^2)$, so that very fine grids usually imply poor convergence. This is the problem that multigrid methods is designed to cure. In this light we examine here the convergence properties of the numerical method implemented in the Proteus code. Computations were performed for the grids used in the published test cases and for two other grids which were either twice as coarse or twice as fine, in each direction.

III.1 Convergence Analysis

The growth rate for each of the above cases was calculated based on the convergence history results obtained from Proteus. In each of the cases, residuals are calculated for the continuity equation, x-momentum equation y-momentum equation. The growth rate of the maximum residual (resmax), average residual (resavg) or the l_2 norm ($resl_2$) was

computed as:

$$Growth\ \ rate = \left(\frac{Residual^{n_2}}{Residual^{n_1}} \right)^{\frac{1}{n_2 - n_1}} \tag{147}$$

where n_1 and n_2 stand for the time levels. The computed growth rates are tabulated in Appendix H for all the cases.

Note that the residuals used in the computation of the growth rate include the artificial viscosity terms.

III.2 Result and Discussion

These results were obtained for the following grids:

Test Case	Run	Flow Problem		Standard	Twice as fine	Twice as coarse
1	1	Flow over a Circular Cylinder	Inviscid	21 x 51	41 x 101	11 x 26
	2		Viscous	51 x 51	101 x 101	26 x 26
2	1	Turbulent Flow over a Flat Plate	Baldwin-Lomax	81 x 51	161 x 101	41 x 26
	2		Chien k-ϵ, CFL = 2	81 x 51	161 x 101	41 x 26
	3		Chien k-ϵ, CFL = 10	81 x 51	161 x 101	41 x 26
3	3	Sajben Transonic Diffuser Flow	Steady case	81 x 51	161 x 101	41 x 26

The computational grids for test case 1, run 1 are shown in Figs. III.1–III.3, and the residual histories are shown in Figs. III.4–III.6. The convergence rates are generally good, but they detoriorate significantly with grid refinement. Corresponding grids and results for the viscous flow are shown in Figs. III.7–III.12. We see the same trends with grid refinement.

The computational grids for the turbulent boudary layer flow over a flat plate are shown

in Figs. III.13–III.15 and the convergence histories are given in Figs. III.16–III.24. The convergence rates are much slower than those of the inviscid (Euler) or laminar viscous flow in test case 1. Again, we see the trend towards poorer convergence rates with grid refinement. however, at long times, the convergence stagnates and the residuals become stuck at levels which are different for each grid. The likely cause is the idiosyncracy of the turbulence model and its treatment of wall boundary conditions. The latter may explain why the coarsest grid computations sometimes got stuck at higher residual levels than the others.

The computational grids for the Sajben transonic flow are shown in Figs. III.25–III.27. The residual histories are given in Figs. III.28–III.30.

CONCLUDING REMARKS

A comprehensive review of the application of multigrid methods to compressible fluid flow computations has been presented. Although the desirable goal of grid-independent convergence rate was not fully realized in most of the computations, speedup in convergence rate by a factor of between 2 to 10 was typically achieved through the use of multigrid procedures. Such results have been obtained with numerical methods similar to those implemented in the Proteus computer code.

Detailed stability analysis of numerical schemes for solving the Euler and Navier-Stokes equations in two- and three-dimensional compressible fluid flow has been presented. The results show the conditions for which good residual smoothing required for effective multigrid convergence can be obtained. Effects of Reynolds number, Mach number, artificial dissipation, and aspect ratio on convergence and smoothing properties are illustrated. Local mode analysis with frozen coefficients can provide information about convergence rates in practical computations.

Actual computations with the Proteus code on grids with different levels of refinement show that the convergence rate deteriorates as the grid is refined. This is the main problem to be cured by the multigrid implementation.

ACKNOWLEDGEMENT

This work is funded by NASA Lewis Research Center under Grant No. NAG-3-1329 with Dr. Jim Scott as Technical Monitor. Part of it was also conducted in residence at ICOMP, NASA Lewis Research Center. Computations were performed on supercomputers at NASA Lewis and NASA Ames Research Centers. We received substantial technical assistance from Mr. A. O. Oloso.

REFERENCES

ANDERSON, D. A., TANNEHILL, J. C. and PLETCHER, R. H., 1984, Computational Fluid Mechanics and Heat Transfer, McGraw Hill, New York.

ANDERSON, W. K., THOMAS, J. L., and WHITFIELD, D. L., 1988, Multigrid Acceleration of the Flux-Split Euler Equations, AIAA J., Vol. 26, No.6, pp. 649-654

ANDERSON, W. K., THOMAS, J. L., and WHITFIELD, D. L., 1988, Three-Dimensional Multigrid Algorithms for the Flux-Split Euler Equations, NASA Technical Paper 2829

ANDERSON, W. K., van LEER, B., 1985, A comparison of Finite Volume Flux Vector Splittings for the Euler Equations, AIAA paper 85-0122.

ARNONE, and SWANSON, R. C., 1988, A Navier-Stokes Solver for Cascade Flows, NASA-CR 181682.

BAKHALOV, N. S., 1966, On the Convergence of a Relaxation Method with Natural Constraints on an Elliptic Operator, Z. Vycisl, Mat. Mat. Fiz. Vol. 6, pp.861-885.

BAYSAL, O., FOULADI, R., and LESSARD, V. R., 1991, Multigrid and Upwind Viscous Flow Solver on Three-Dimensional Overlapped and Embedded Grids, AIAA J., Vol. 29, pp. 903-910.

BEAM, R. M. and WARMING, R. F., 1978, An Implicit Scheme for the Compressible Navier-Stokes Equations, AIAA J., Vol. 16, pp.393-402

BRANDT, A. 1977, Multi-level Adaptive Solutions to Boundary-Value Problems, Math. Comp. Vol. 31, No. 138, pp. 330-390.

CAUGHEY, A.D., 1988, Diagonal Implicit Multigrid Algorithm for the Euler Equations, AIAA J., Vol. 26, pp. 841-851

CAUGHEY, A. D. and IYER, 1989, Diagonal Implicit Multigrid Calculation of Inlet Flowfield, AIAA J., Vol. 27, pp. 110–112.

CHIMA, E., TURKEL, E., and SCHAFFER, S., 1987, Comparison of Three Explicit Multigrid Methods for the Euler and Navier-Stokes Equations. (NASA TM 88878).

DEMUREN, A. O., 1989, Application of Multigrid Methods for Solving the Navier-Stokes Equations, Proc. Ins. Mech. Eng., J. Mech. Eng. Sci., Vol. 203 (Also NASA TM 102359).

DEMUREN, A. O., 1992, Multigrid Acceleration and Turbulence Models for Computations of 3–D Turbulent Jets in Crossflow, Int. J. Heat and Mass Transfer, in press.

FEDERENKO, R. P., 1961, A Relaxation Method for Solving Elliptic Difference Equations, Z. Vycisl, Mat. Mat. Fiz., Vol. 1, pp. 922-927.

FEDERENKO, R. P., 1964, On the Speed of an Iteration Process, Z. Vycisl, Mat. Mat. Fiz., Vol. 1, pp. 922-927.

HOFFMANN, K. A., 1989, Computational Fluid Dynamics for Engineers, Engineering Education System, Texas.

HIMANSU, and RUBIN, 1988, Multigrid Acceleration of a Relaxation Procedure for the Reduced Navier-Stokes Equations, AIAA J., Vol. 26, pp.1044–1051.

HIRSCH, C., 1988, Numerical Computation of Internal and External Flows, Vol. 1 & 2,John Willey and Sons, New York.

JAMESON, A., 1979, Acceleration of Transonic Potential Flow Calculations on Arbitrary Meshes by the Multiple grid Method, AIAA Paper 79–1458.

JAMESON, A., 1983, Solution of the Euler Equations for Two-Dimensional Transonic Flow by a Multigrid Method, Appl. Math. Comp. Vol. 13, pp.327-355.

JAMESON, A. and YOON, S, (1985), Multigrid Solution of the Euler Equations Using Implicit Schemes, AIAA paper 85-0293.

JAMESON, A. and YOON, S, (1986), Multigrid Solution of the Euler Equations Using Implicit Schemes, AIAA J. Vol. 24, pp.1737-1743.

JAMESON, A. and YOON, S, (1987), Lower-Upper Implicit Schemes with Multiple Grids for the Euler Equations, AIAA J. Vol. 25, pp.929-935.

JESPERSON, D. C., 1983, Design and Implementation of a Multigrid Code for the Euler Equations, J. of Applied Mathematics and Computation, Vol. 13, pp. 357-374

JESPERSON, D. C. and PULLIAM, T.H., 1983, Flux Vector Splitting and Approximate Newton Methods, AIAA Sixth CFD Conference

KOREN, B., 1990, Multigrid and Defect Correction for the Steady Navier-Stokes Equations, J. Comp. Phys., Vol. 87, pp.25-46.

MAVRIPLIS, D. J., 1988, Multigrid Solution of the Two-Dimemsional Euler Equations on Unstructured Triangular Meshes, AIAA J., Vol. 26, pp. 824-831

MAVRIPLIS, D. J., 1990, Accurate Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes, AIAA J., Vol. 28, pp. 213-221.

MAVRIPLIS, D. J. and JAMESON, A., 1990, Multigrid Solution of the Navier-Stokes Equations on Triangular Meshes, AIAA J., Vol. 28, pp.1415-1425.

McCARTHY, D.R., and REYHNER, T.A., 1982, Multigrid Code for Three Dimensional Transonic Flows, AIAA J., Vol. 20, pp.45-53.

MULDER, W. A., 1989, A New Multigrid Approach to Convection Problems, J. Comp. Phy., Vol. 83, pp.303-323.

NI, R. H., 1981, A Multiple Grid Scheme for Solving the Euler Equations, AIAA J., Vol. 20, pp.1565-1571.

PULLIAM, T. H., 1986, Artificial Dissipation Models for the Euler Equations, AIAA J., Vol. 24, pp. 1931-1940.

RADESPIEL, R., ROSSOW, C. and SWANSON, R. C., 1990, Efficient Cell-Vertex Multigrid Scheme for Three-Dimensional Navier-Stokes Equtions, AIAA J., Vol. 28, pp.1464-1472.

RHIE, C. M., 1989, Pressure-Based Navier-Stokes Solver Using Multigrid Method, AIAA J., Vol. 27, pp. 1017.

SOUTH, J. C. and BRANDT, A., 1976, The Multi-Grid Method : Fast Relaxation for Transonic Flows, Adv. Eng. Sci. Vol. 4, pp.1359-1369.

SOUTHWELL, R. V., 1935, Stress-Calculation in Frameworks by the Method of "Systematic Relaxation of Constraints", Proc. R. Soc. London, Vol. 151A, No. 872, 56-95.

STEGER, J.L., and WARMING, R.F., 1981, Flux Vector Splitting of the Inviscid Gas Dynamic Equation with Application to Finite Difference Methods, J. Comp. Phys., Vol. 40, pp. 263-293.

STIEFEL, E. L., 1952, Uber einige Methodern der Relaxationsrechnung, Z.A.M.P., Vol. 3, pp. 1-133

SWANSON, R. C. and RADESPIEL, R., 1991, Cell Centered and Cell Vertex Multigrid Schemes for Navier-Stokes Equations, AIAA J., Vol. 29, pp. 697.

SWANSON, R. C. and TURKEL, E., 1987, Artificial Dissipation and Central Difference Schemes for the Euler and Navier-Stokes Equations, AIAA paper 87-1107

THOMPSON, M. C. and FERZIGER, J. H., 1989, An Adaptive Multigrid Techninique for the Incompressible Navier-Stokes Equations, J. Comp. Phys., Vol. 82, pp.94-121.

TONG, S. S., 1987, The Impact of Smoothing Formulations on the Stability and Accuracy of various Time Marching Schemes, AIAA paper 87-1106

VANKA, S. P., 1986, Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables, J. Comp. Phys., Vol. 65, No. 1, pp. 138-158.

von LAVANTE, E., CLAES, D. and ANDERSON, W. K., 1986, The Effect of various Implicit Operators on a Flux Vector Splitting Method, AIAA paper 86-027

YADLIN, Y. and CAUGHEY, A. D., 1991, Block Multigrid Implicit Solution of the Euler Equations of Compressible Fluid Flow, AIAA J., Vol. 29, pp. 712.

YOKOTA, J. W., (1989), A Diagonally Inverted LU Implicit Multigrid Scheme for the 3-D Navier-Stokes Equations and a Two Equation Model of Turbulence, AIAA Paper No. AIAA-89-0

YOKOTA, J. W. and CAUGHEY, A.D., 1988, LU Implicit Multigrid Algorithm for the Three-Dimensional Euler Equations, AIAA J., Vol. 26, pp. 1061.

YOKOTA, J. W., CAUGHEY, A.D., and CHIMA, R.V.,1988, A Diagonally Inverted LU Implicit Multigrid Scheme. (NASA TM 100911).

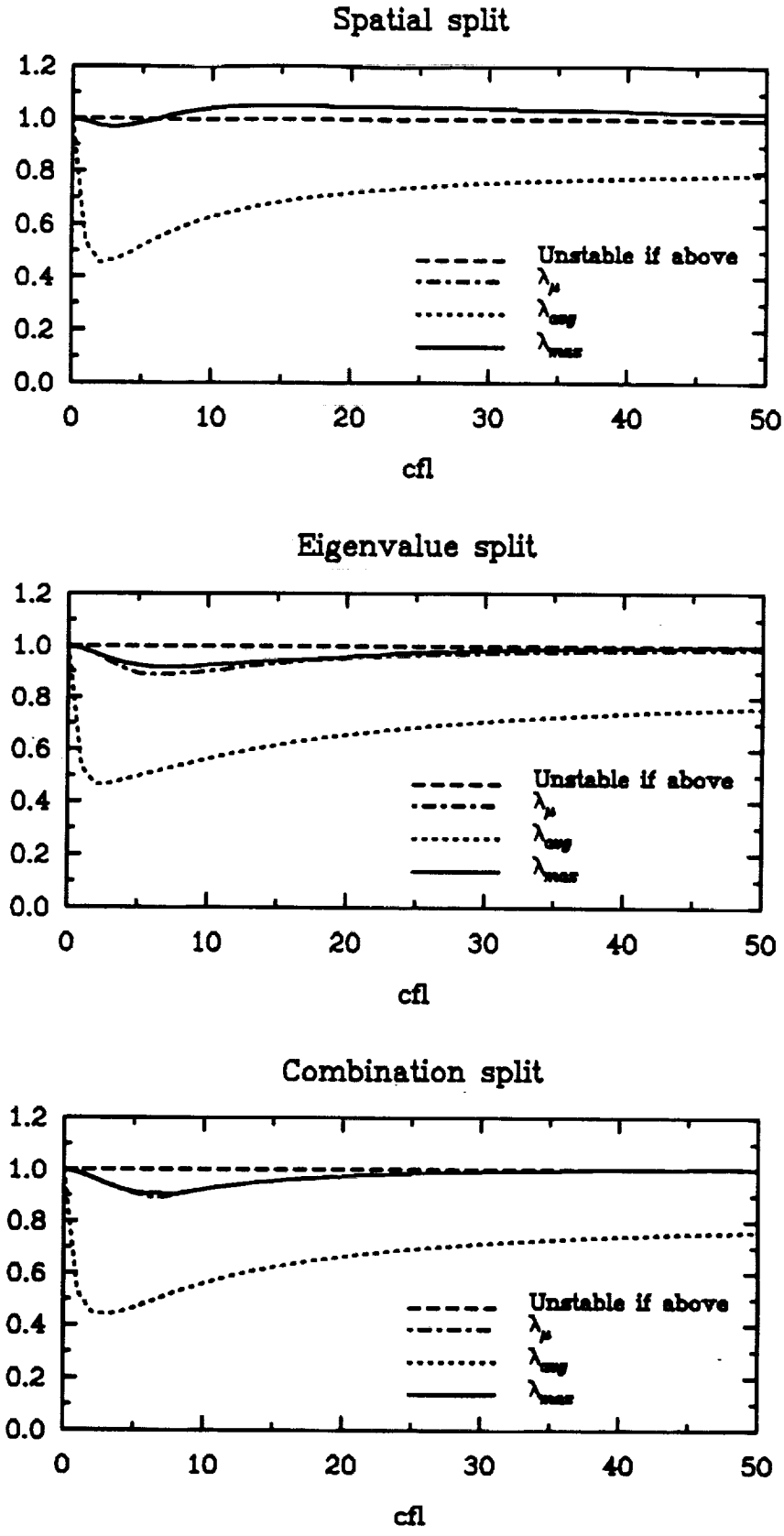


Fig. II.1.1 : Stability Analysis for 3–D Euler Equations (Upwind Diff., Steger & Warming Splitting)

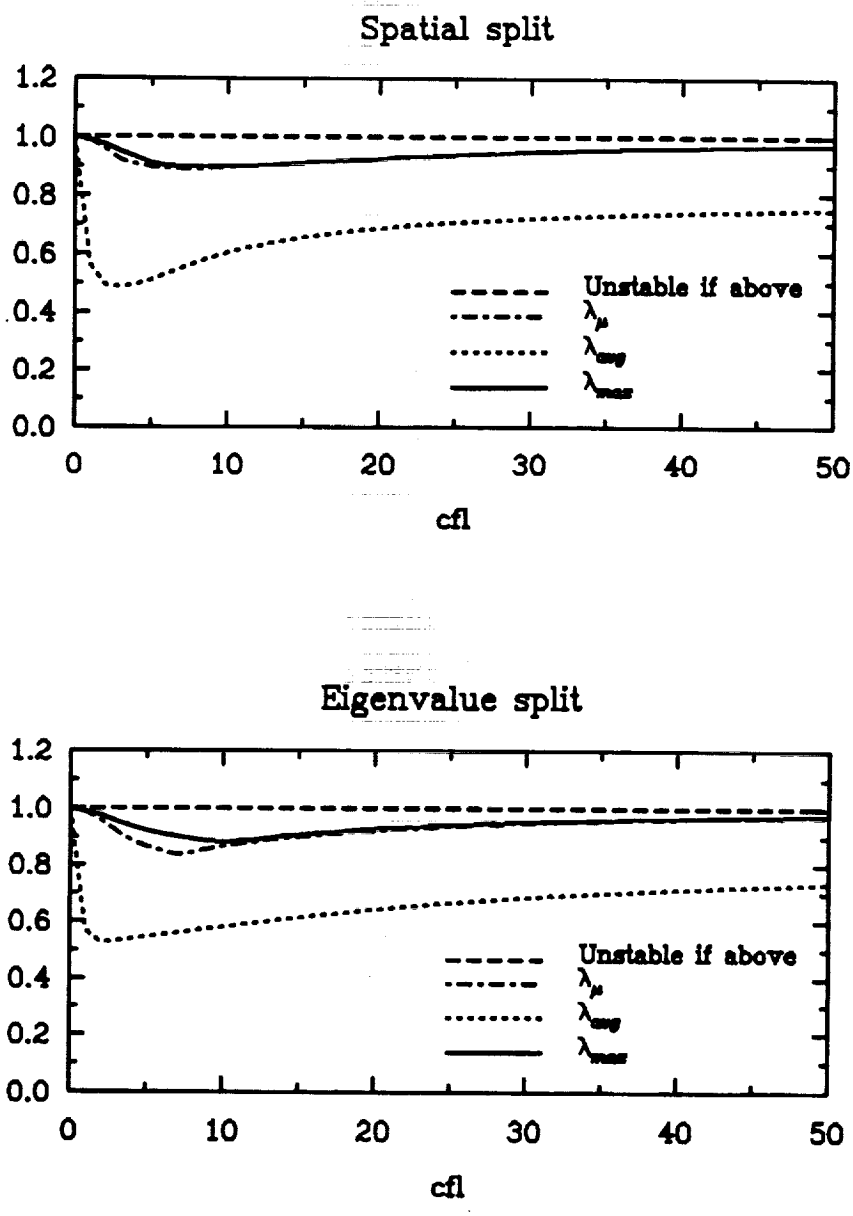


Fig. II.1.2 : Stability Analysis for 2–D Euler Equations (Upwind Diff., Steger & Warming Splitting)

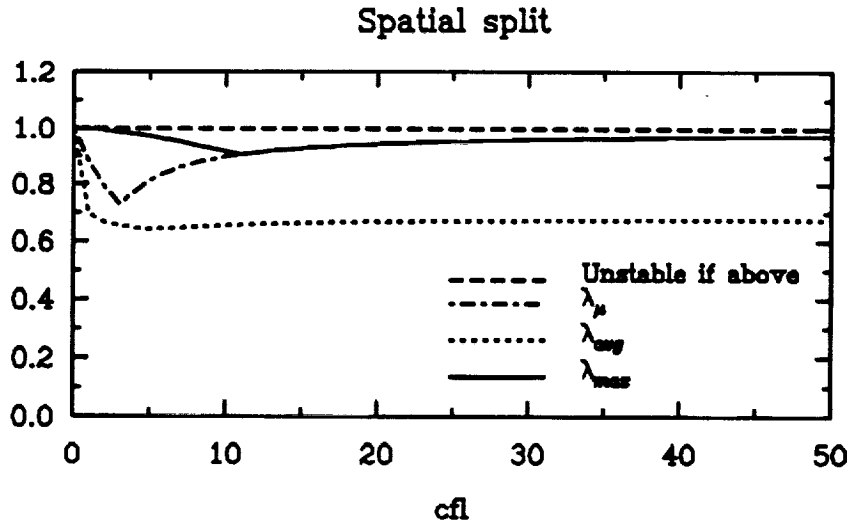


Fig. II.1.3 : Stability Analysis for 1-D Euler Equations (Upwind Diff., Steger & Warming Splitting)

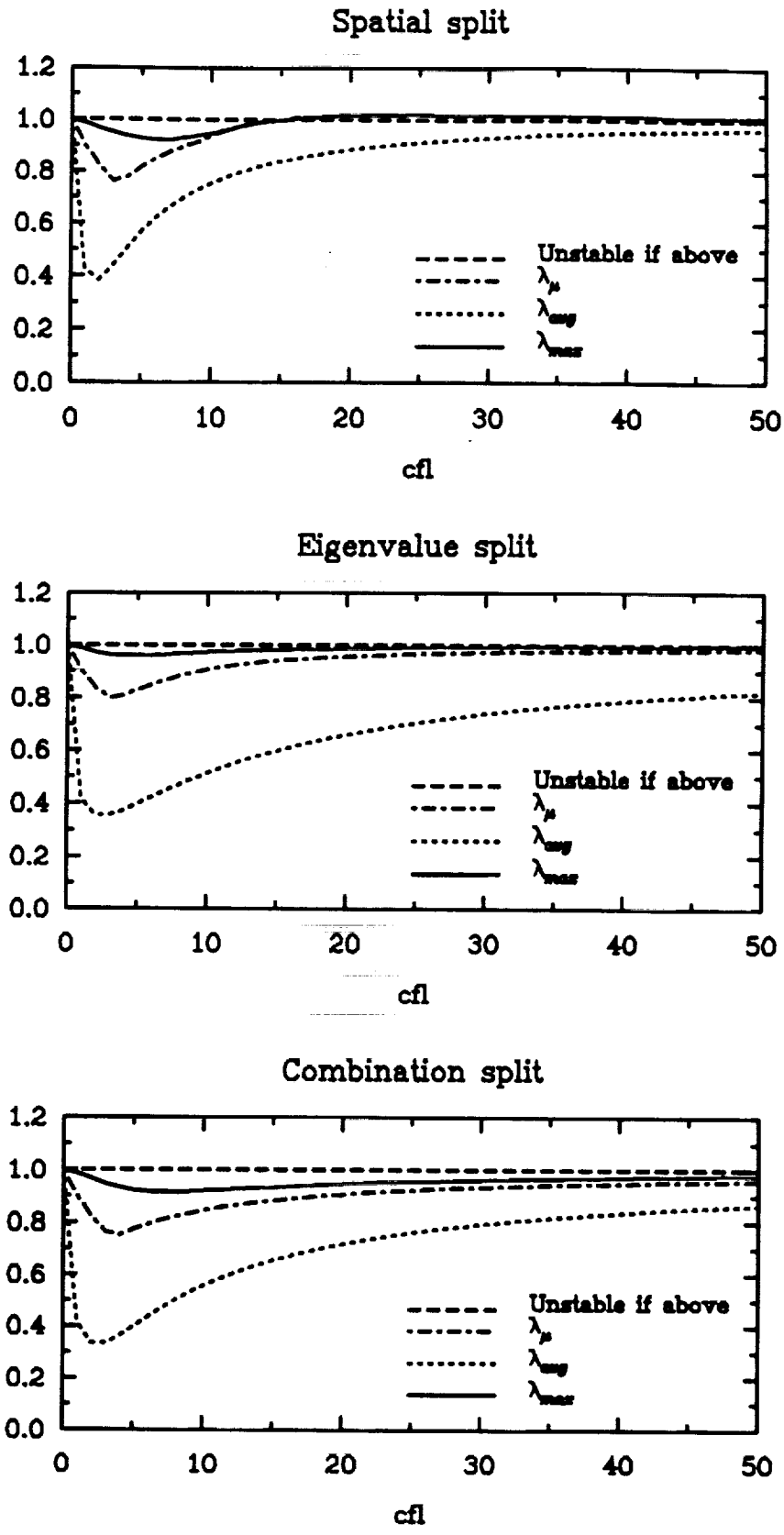
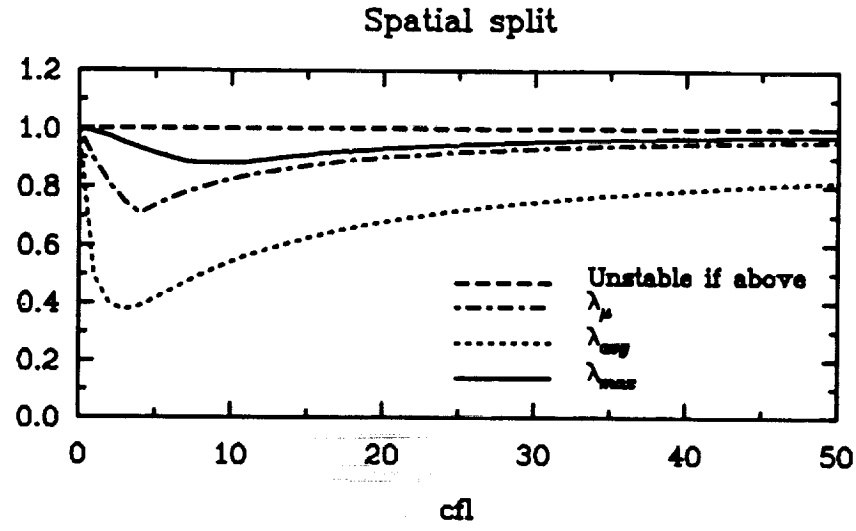


Fig. II.1.4 : Stability Analysis for 3–D Euler Equations (Upwind Diff., Van Leer Splitting)



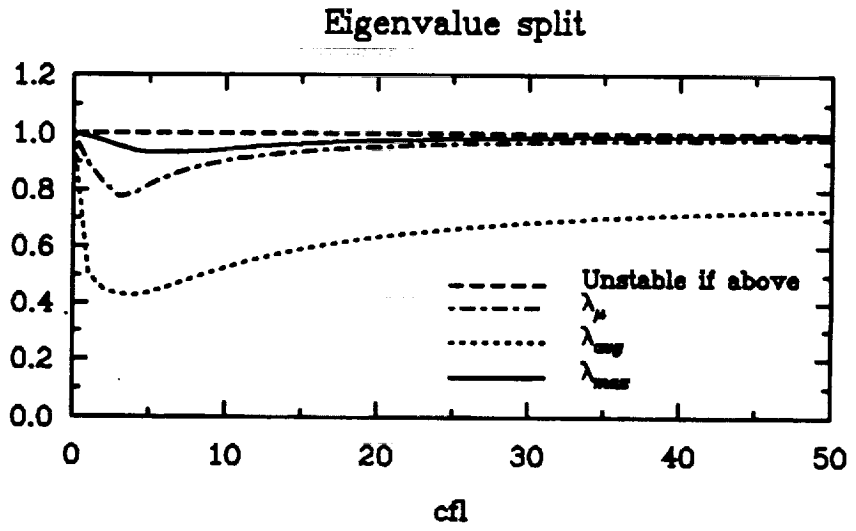


Fig. II.1.5 : Stability Analysis for 2–D Euler Equations (Upwind Diff., Van Leer Splitting)

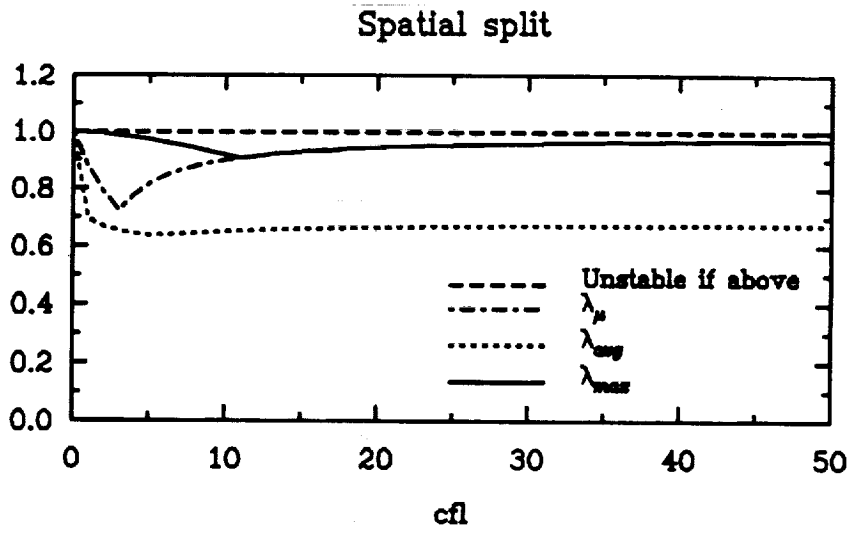


Fig. II.1.6 : Stability Analysis for 1–D Euler Equations (Upwind Diff., Van Leer Splitting)

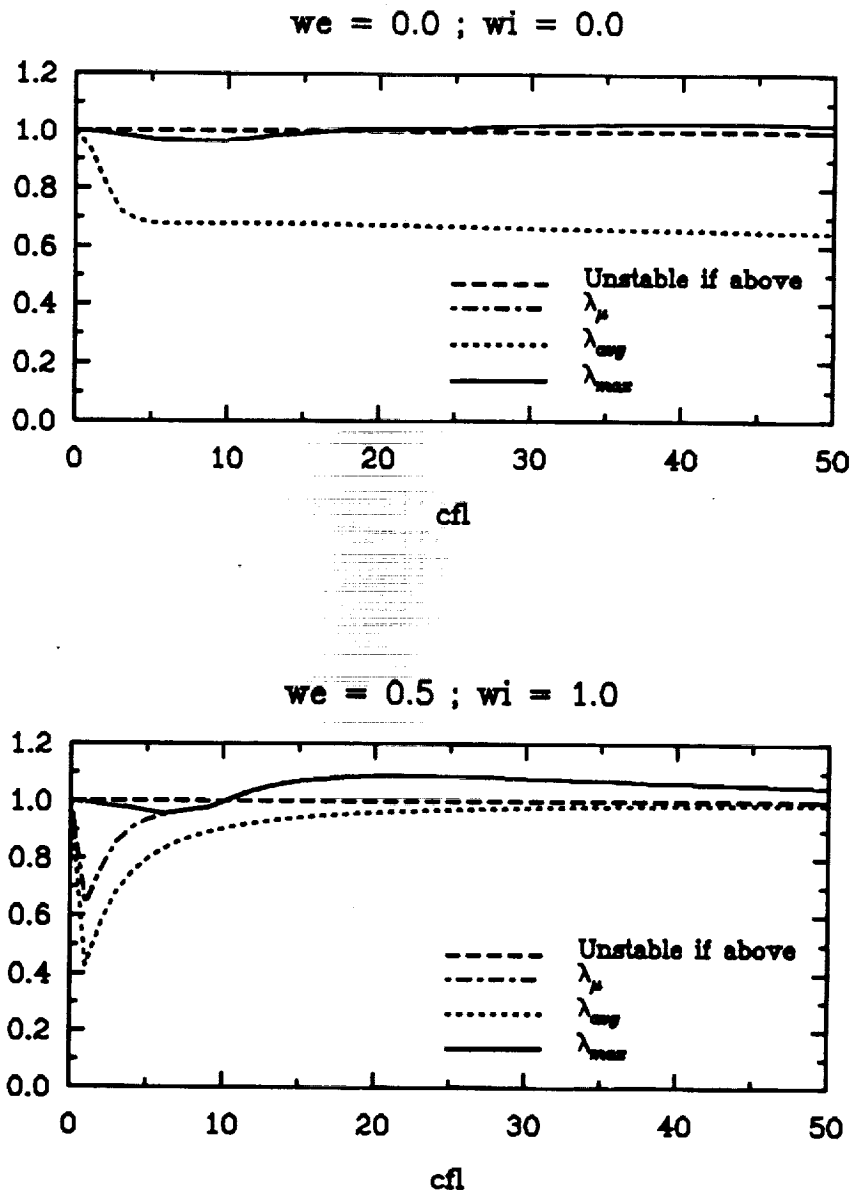
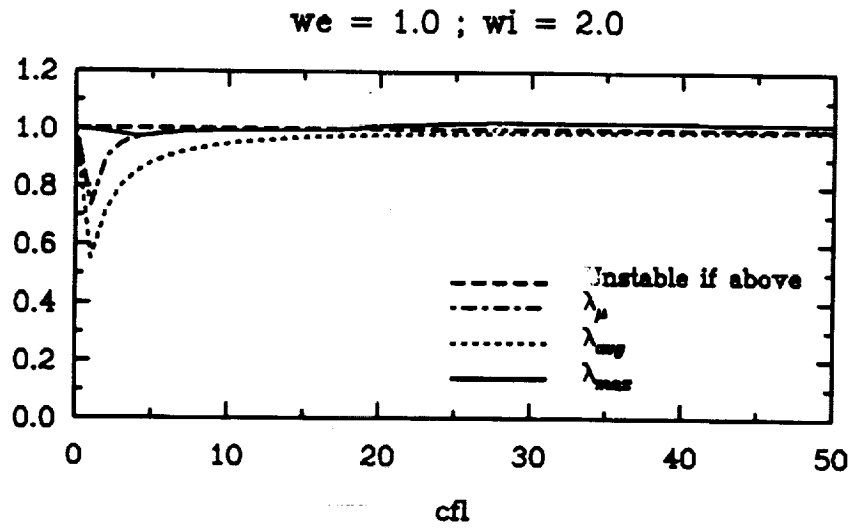


Fig. II.2.1 : Stability Analysis for 3–D Euler Equations (Central Diff.)



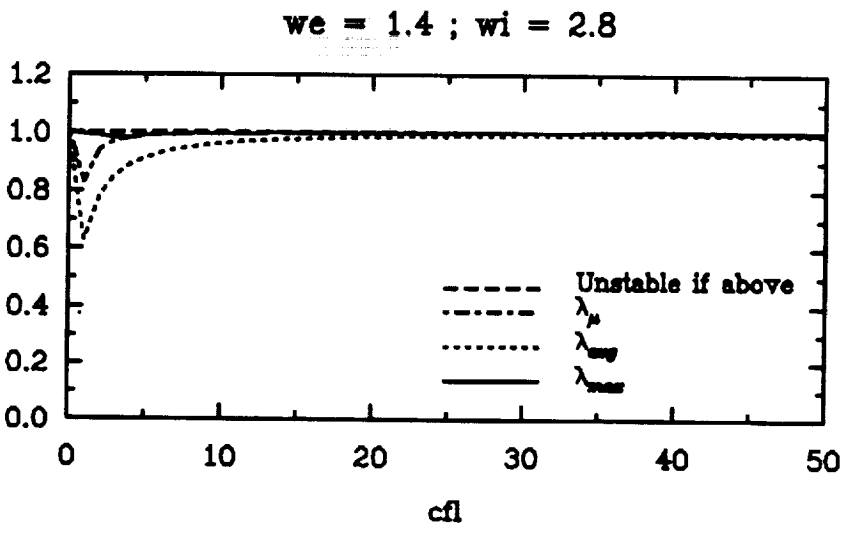


Fig. II.2.2 : Stability Analysis for 3–D Euler Equations (Central Diff.)

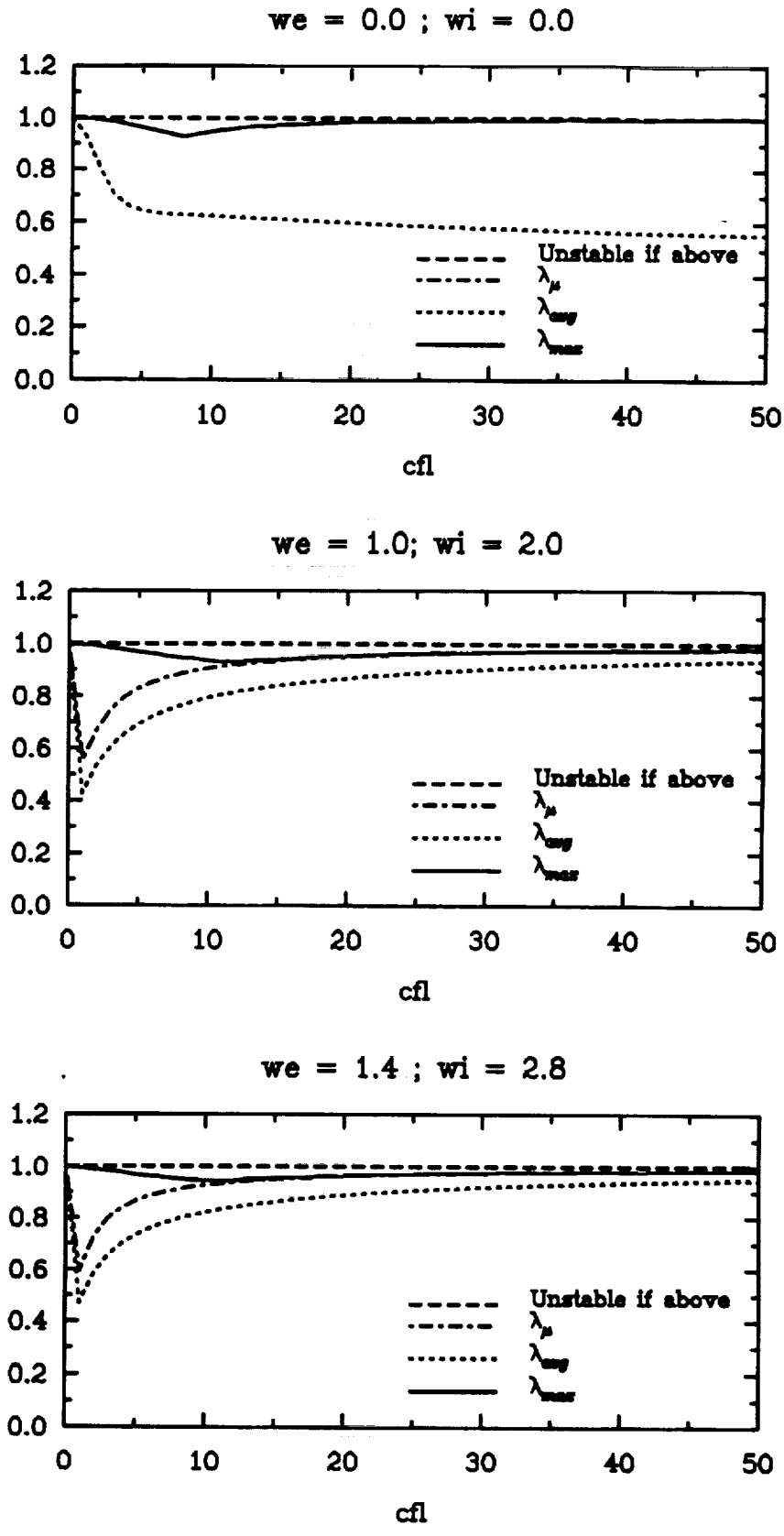


Fig. II.2.3 : Stability Analysis for 2–D Euler Equations (Central Diff.)

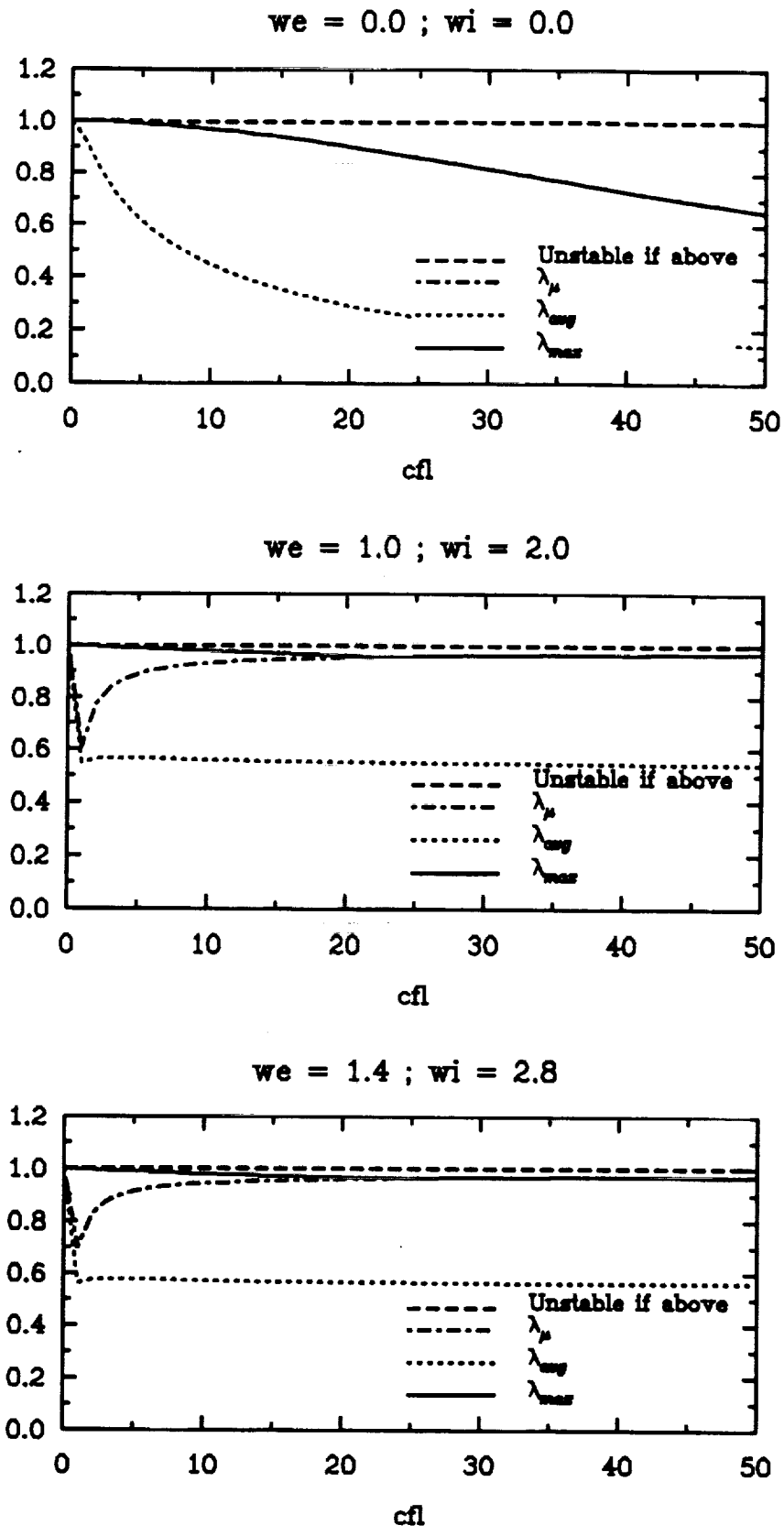
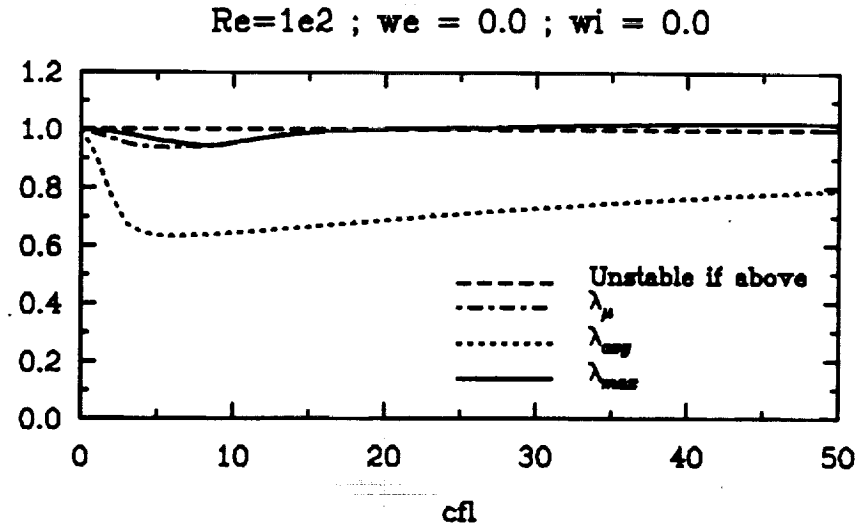


Fig. II.2.4 : Stability Analysis for 1–D Euler Equations (Cent. Diff.)



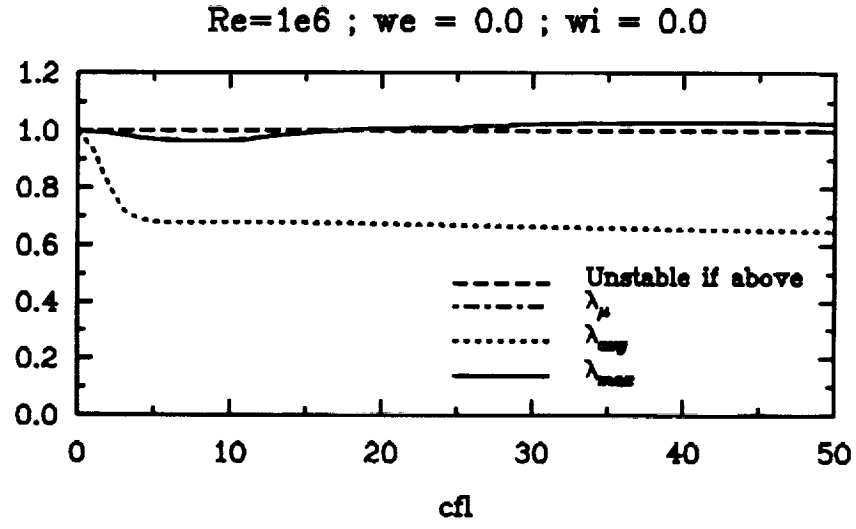
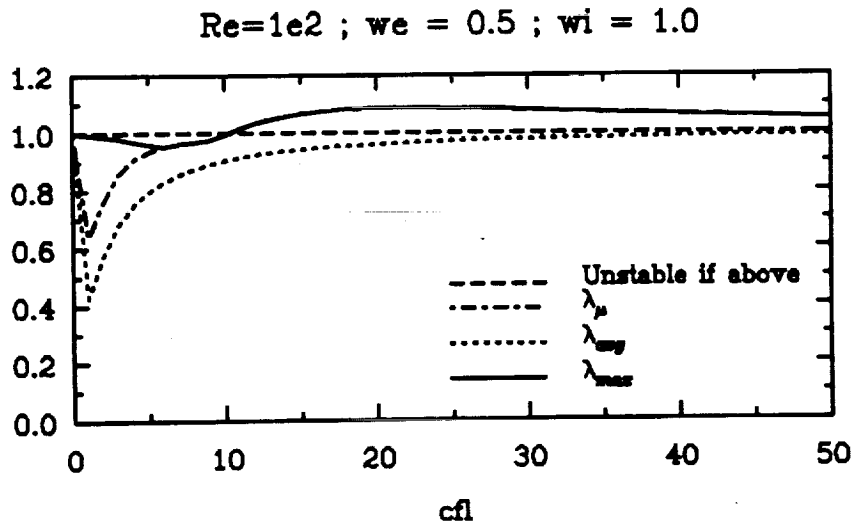


Fig. II.3.1 : Stability Analysis for 3–D Navier–Stokes Equations (Cent. Diff.)



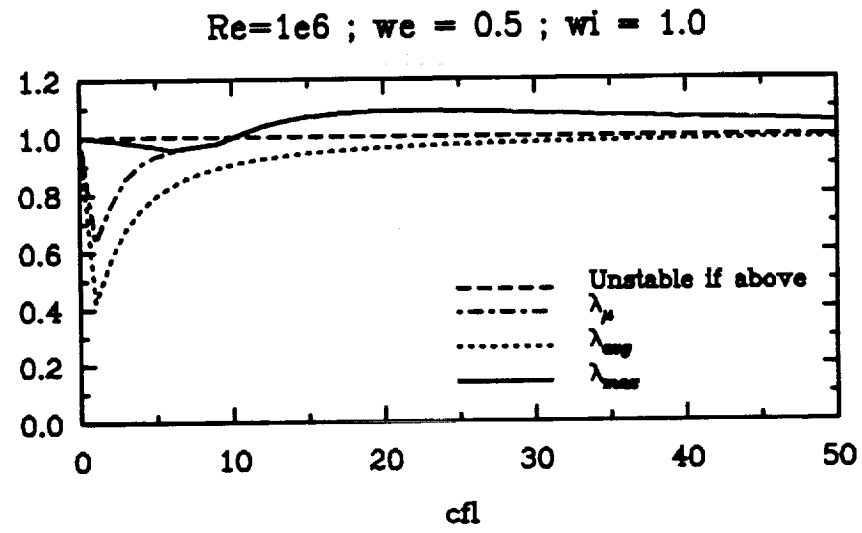
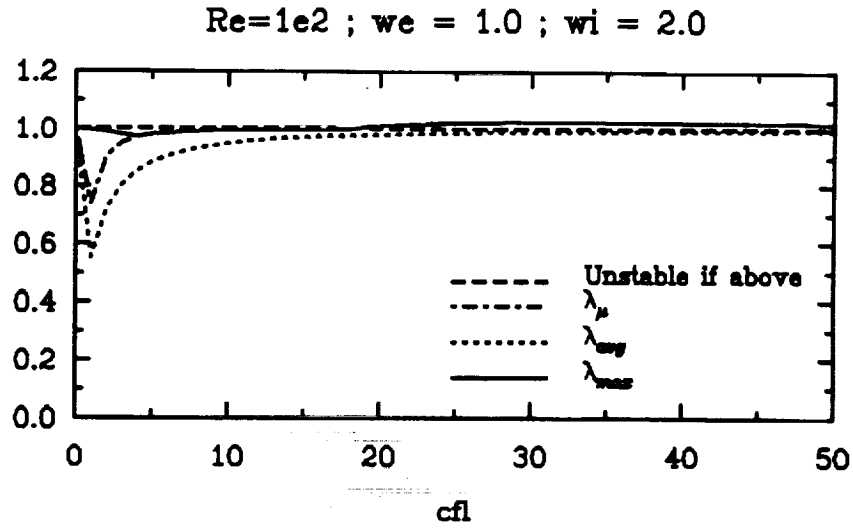


Fig. II.3.2 : Stability Analysis for 3–D Navier–Stokes Equations (Cent. Diff.)



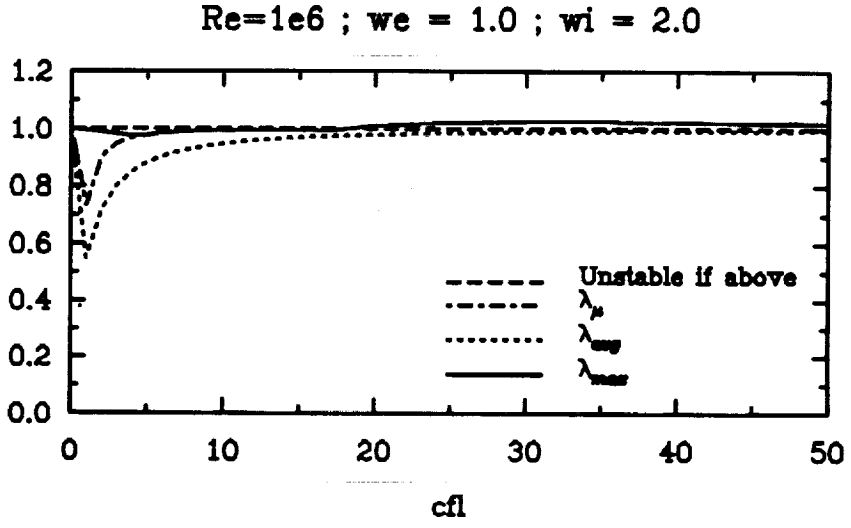
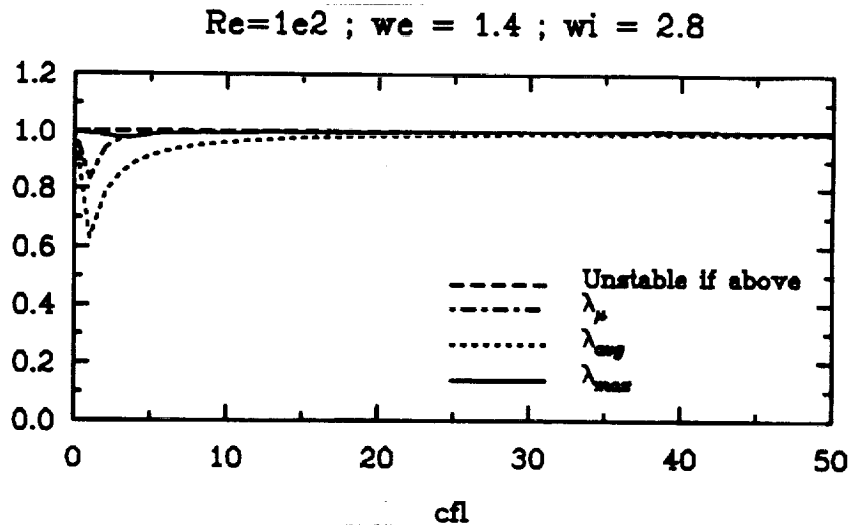


Fig. II.3.3 : Stability Analysis for 3–D Navier–Stokes Equations (Cent. Diff.)



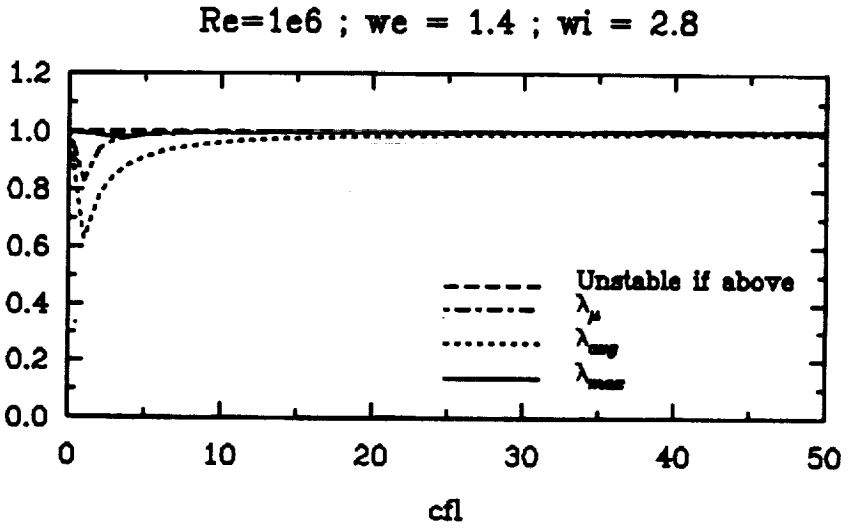
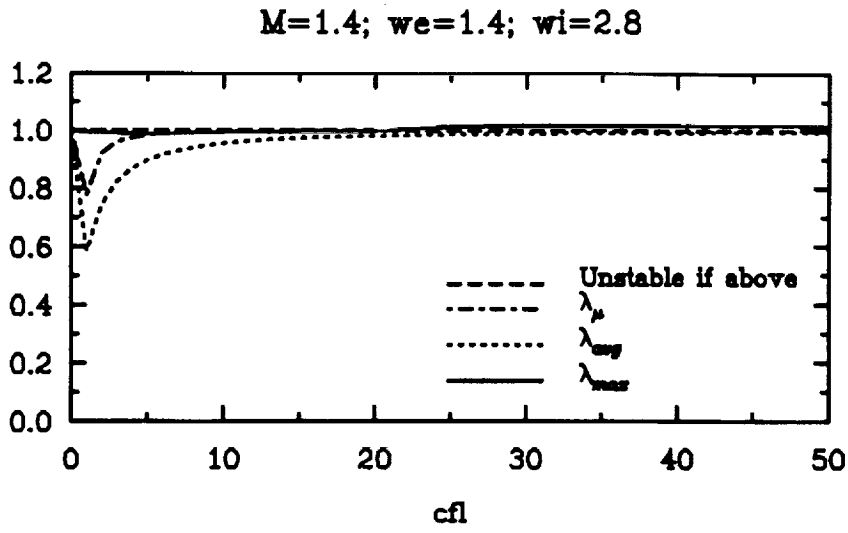


Fig. II.3.4 : Stability Analysis for 3–D Navier–Stokes Equations (Cent. Diff.)



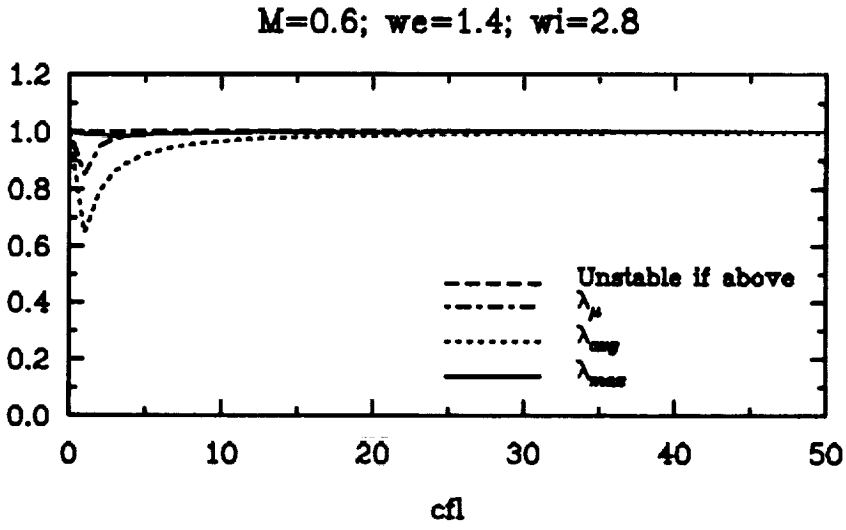
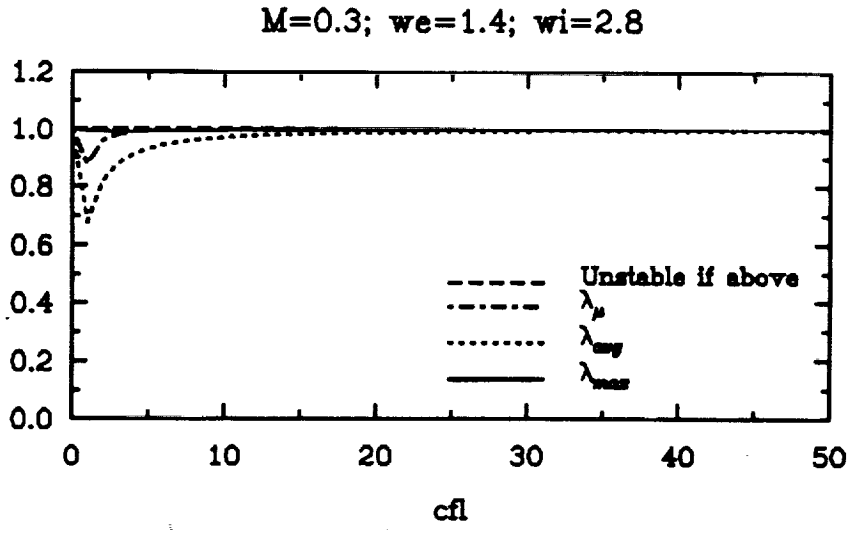


Fig. II.3.5 : Stability Analysis for 3-D Navier-Stokes Equations (Cent. Diff.) Re = 100



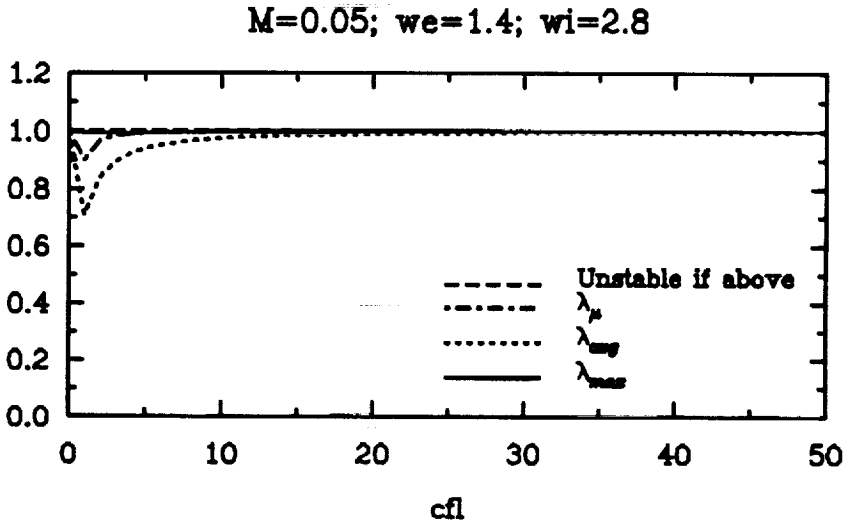
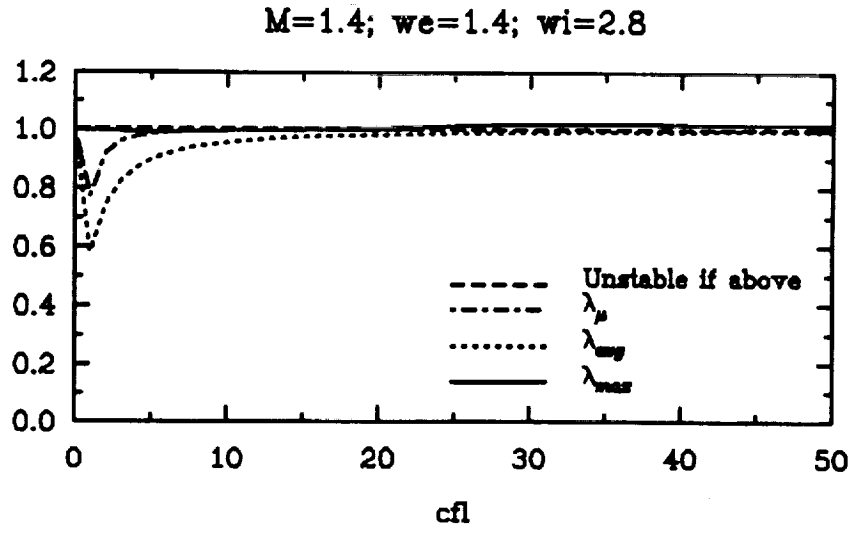


Fig. II.3.6 : Stability Analysis for 3–D Navier–Stokes Equations (Cent. Diff.) Re = 100



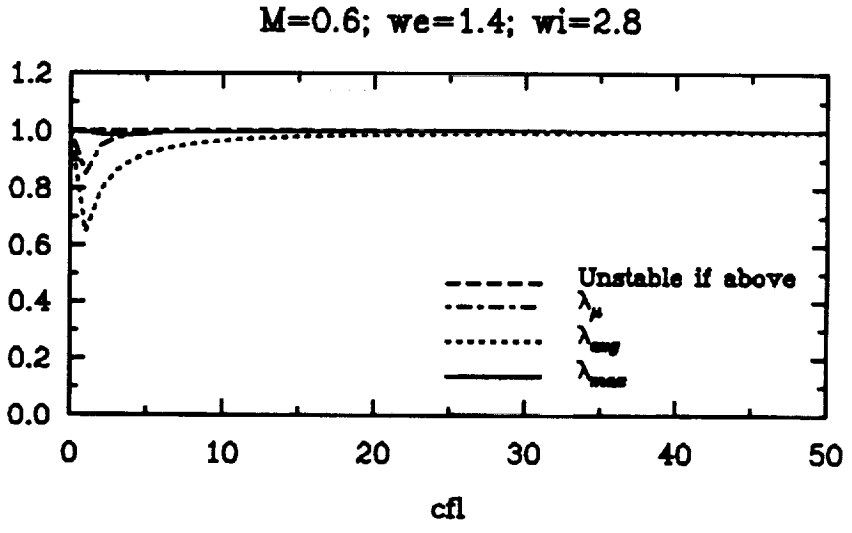
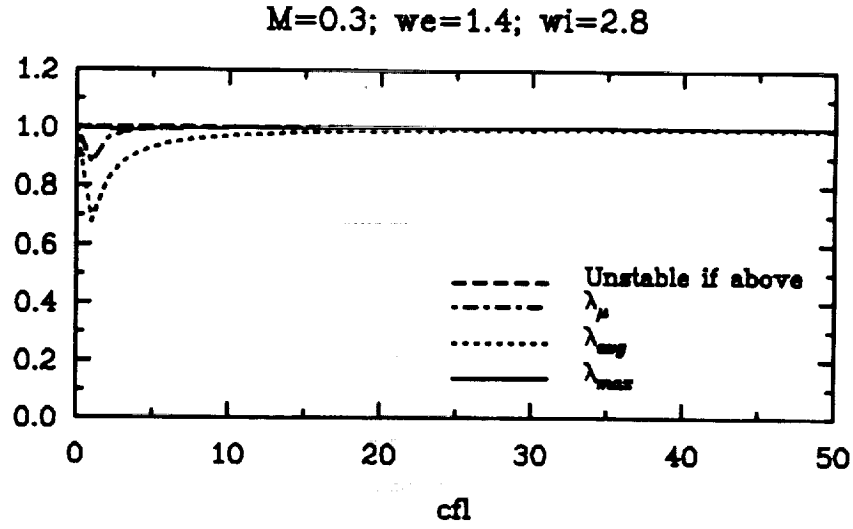


Fig. II.3.7 : Stability Analysis for 3-D Navier-Stokes Equations (Cent. Diff.)
Re = 1000000



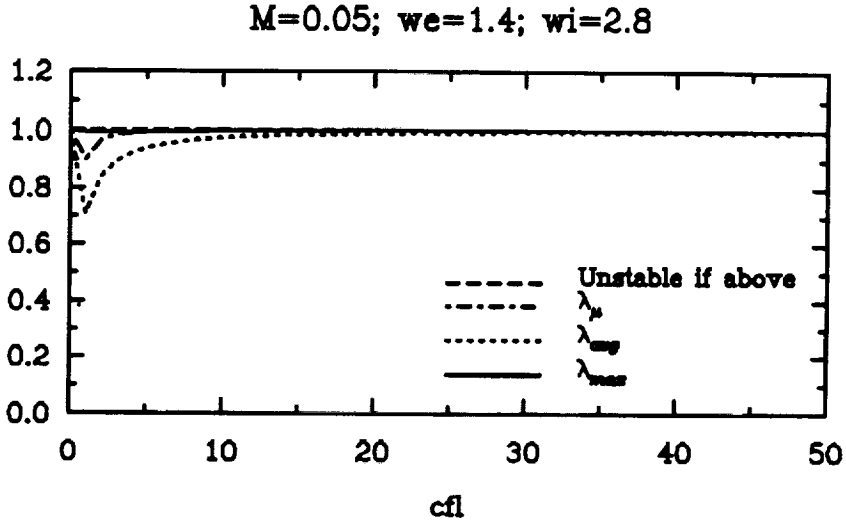
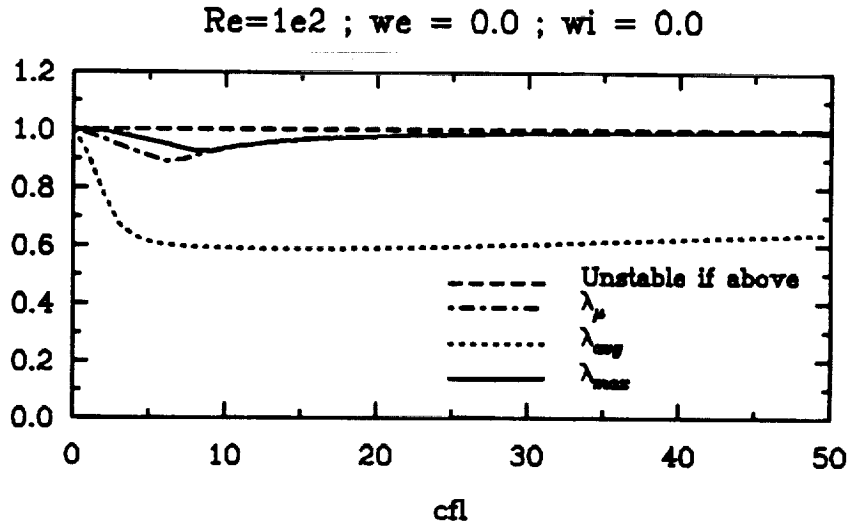


Fig. II.3.8 : Stability Analysis for 3–D Navier–Stokes Equations (Cent. Diff.)
Re = 1000000



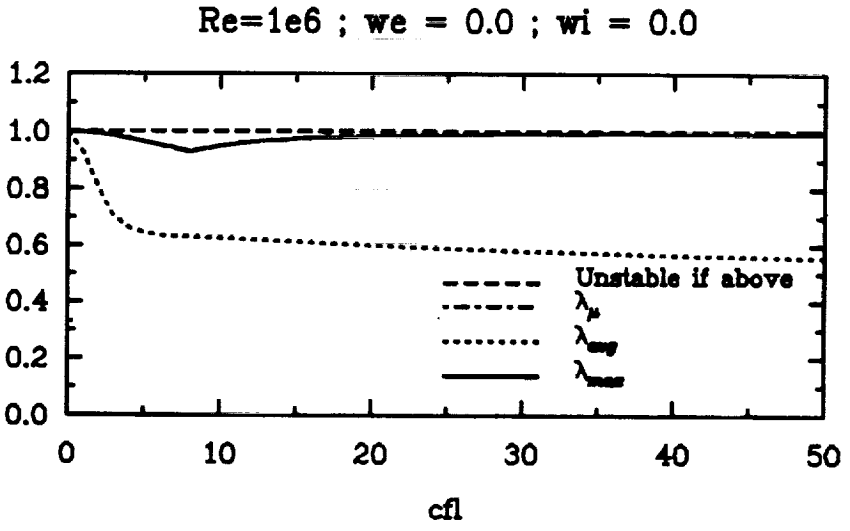
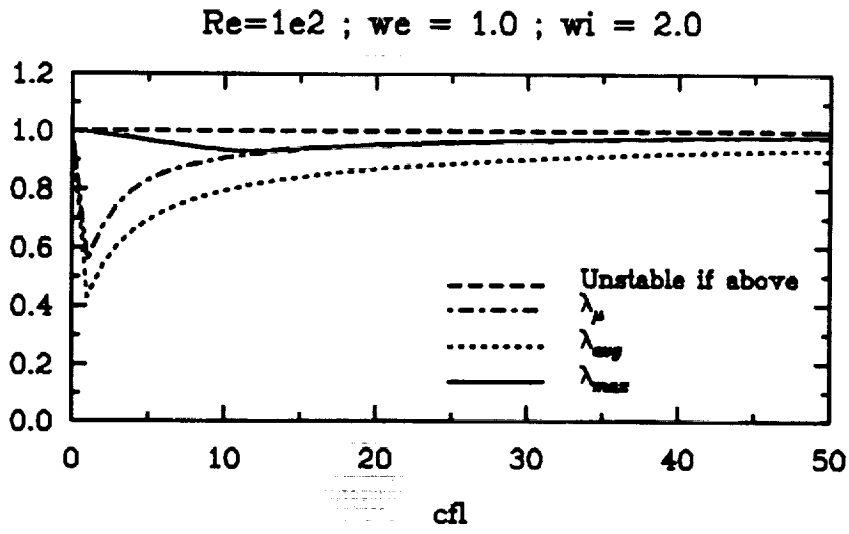


Fig. II.3.9 : Stability Analysis for 2–D Navier–Stokes Equations (Cent. Diff.)



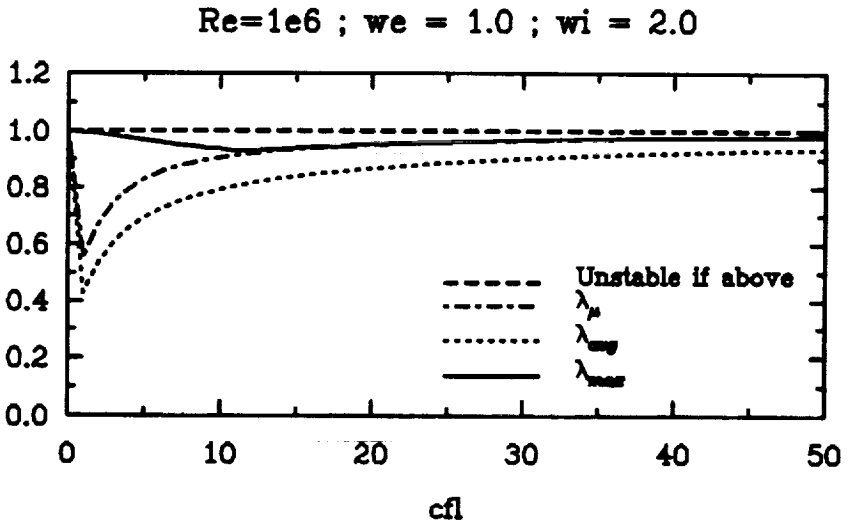


Fig. II.3.10 : Stability Analysis for 2–D Navier–Stokes Equations (Cent. Diff.)

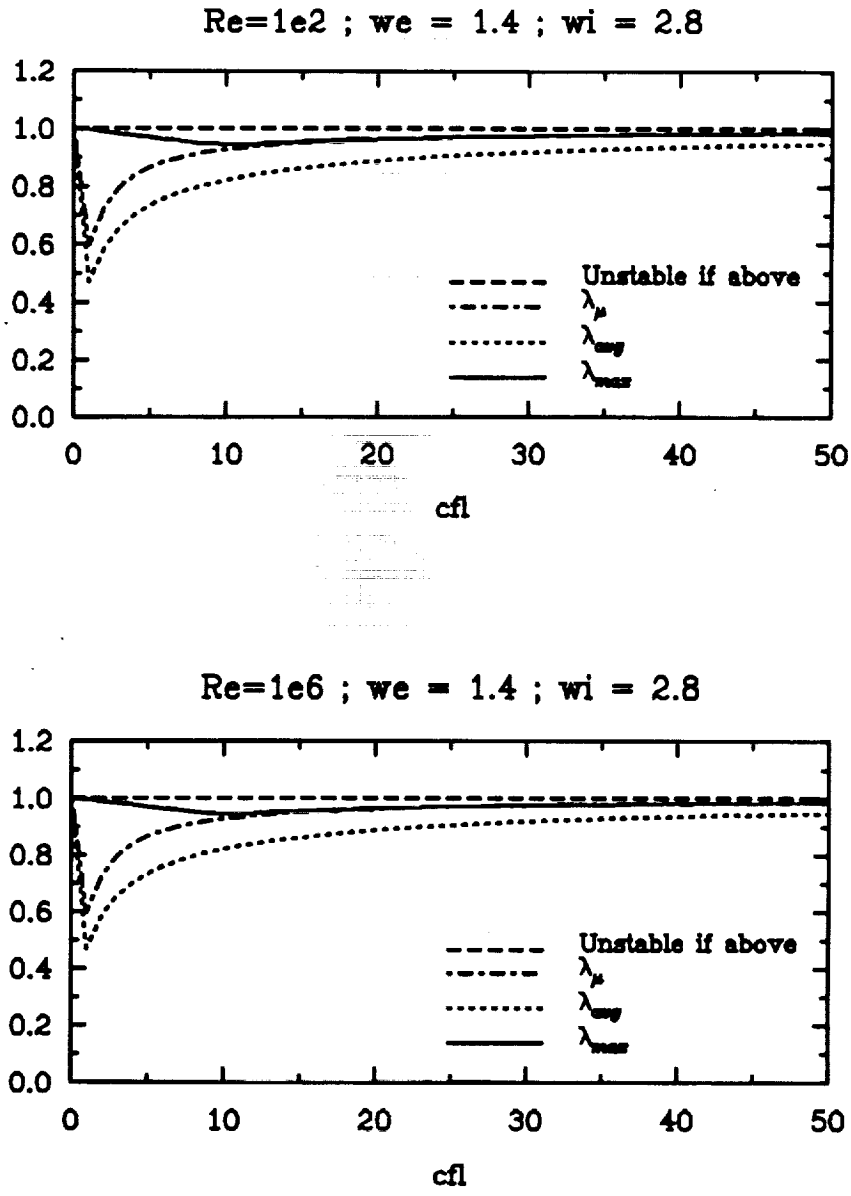
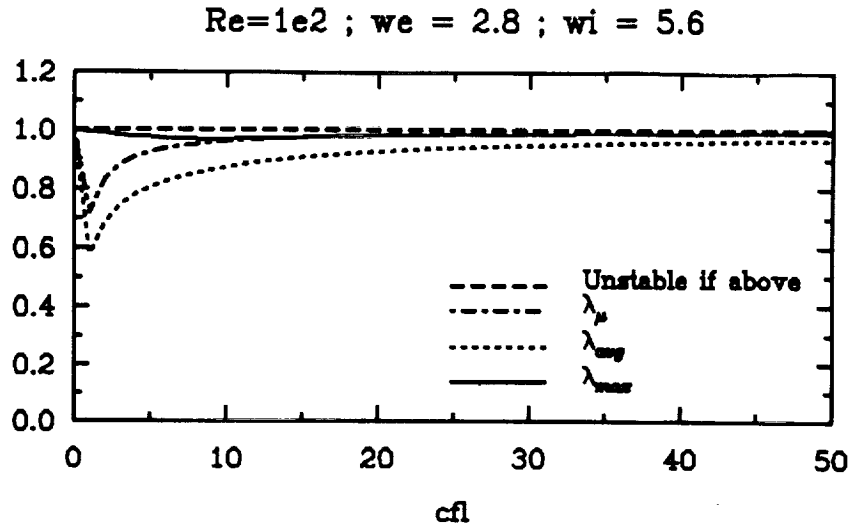


Fig. II.3.11 : Stability Analysis for 2–D Navier–Stokes Equations (Cent. Diff.)



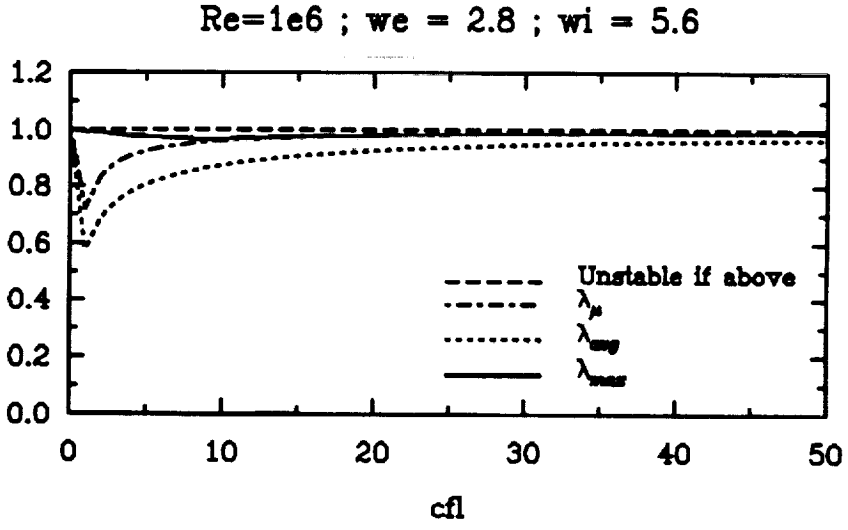
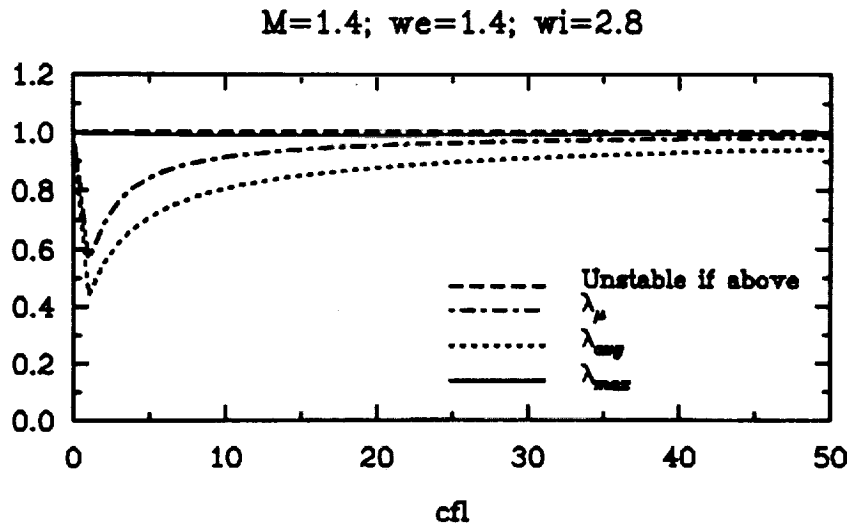


Fig. II.3.12 : Stability Analysis for 2-D Navier-Stokes Equations (Cent. Diff.)



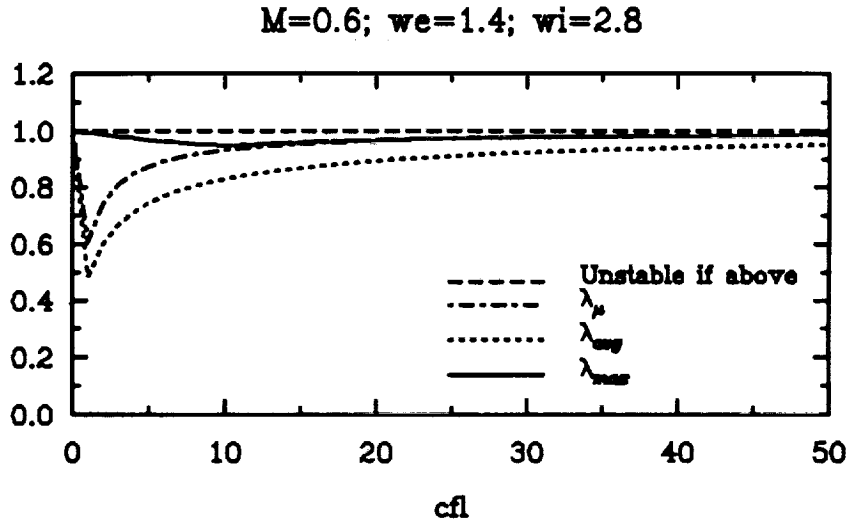
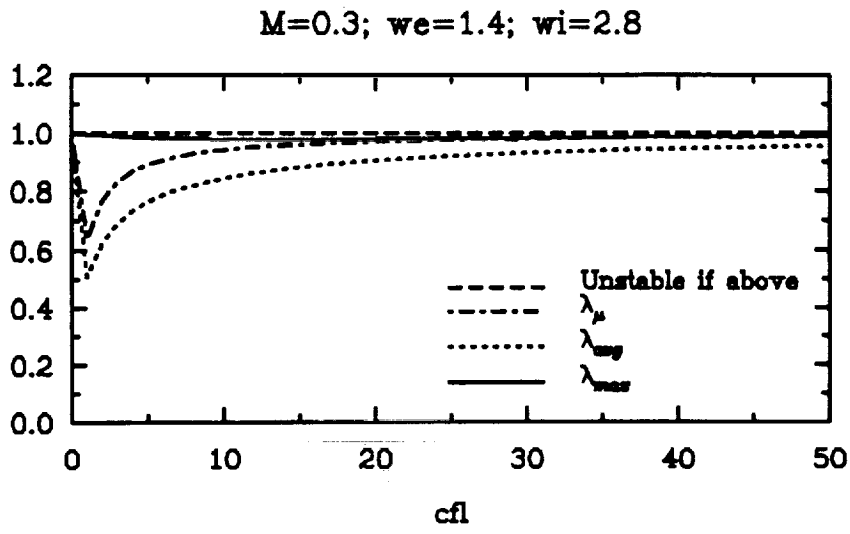


Fig. II.3.13 : Stability Analysis for 2–D Navier–Stokes Equations (Cent. Diff.) Re=100



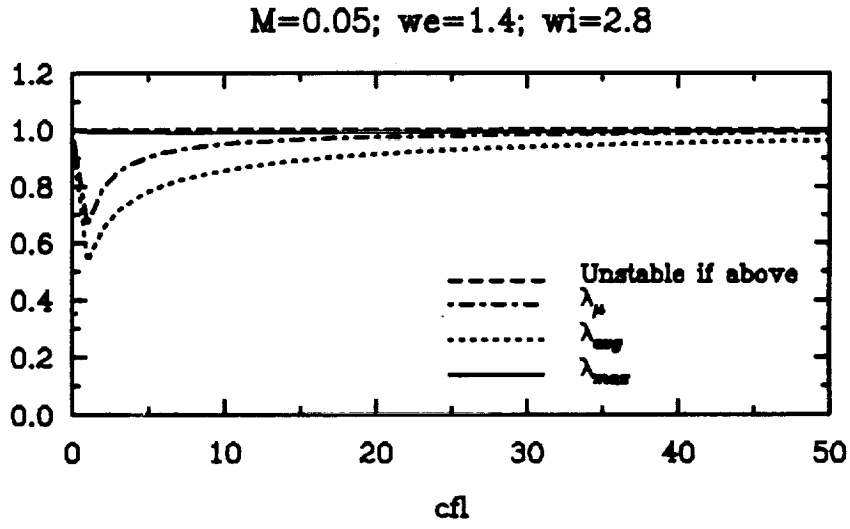
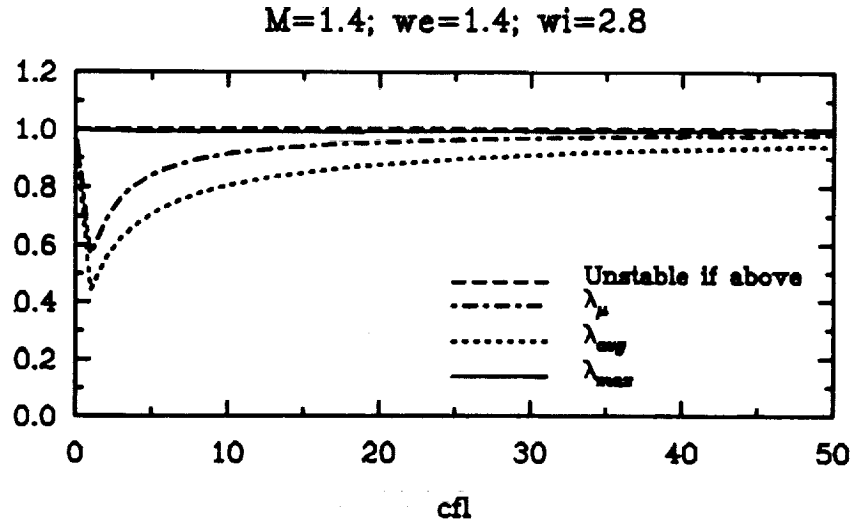


Fig. II.3.14 : Stability Analysis for 2-D Navier-Stokes Equations (Cent. Diff.) Re=100



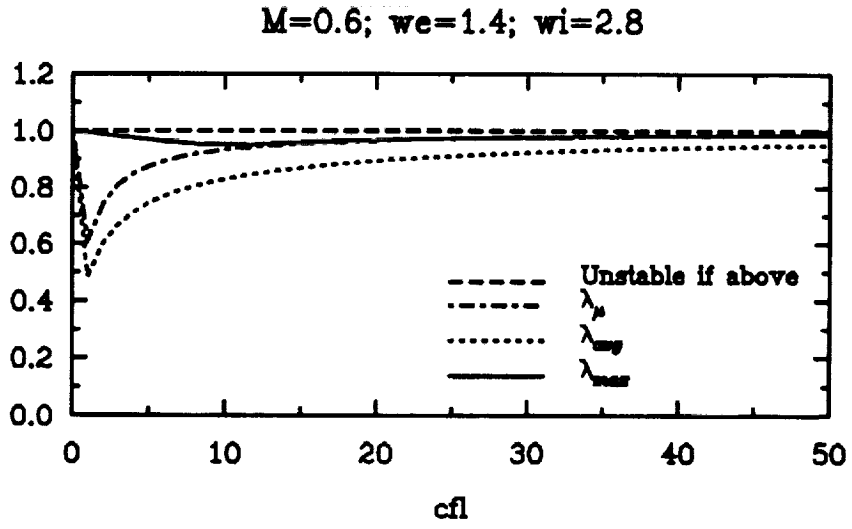
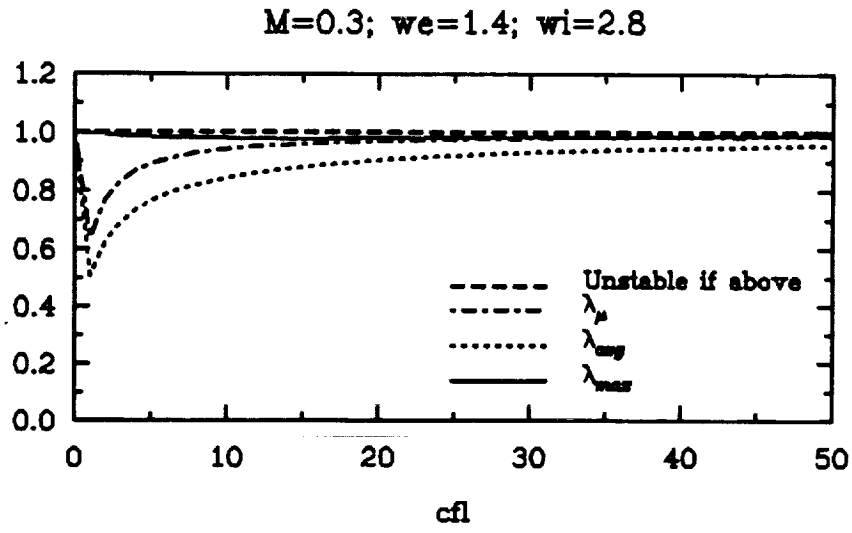


Fig. II.3.15 : Stability Analysis for 2–D Navier–Stokes Equations (Cent. Diff.) Re=1000000



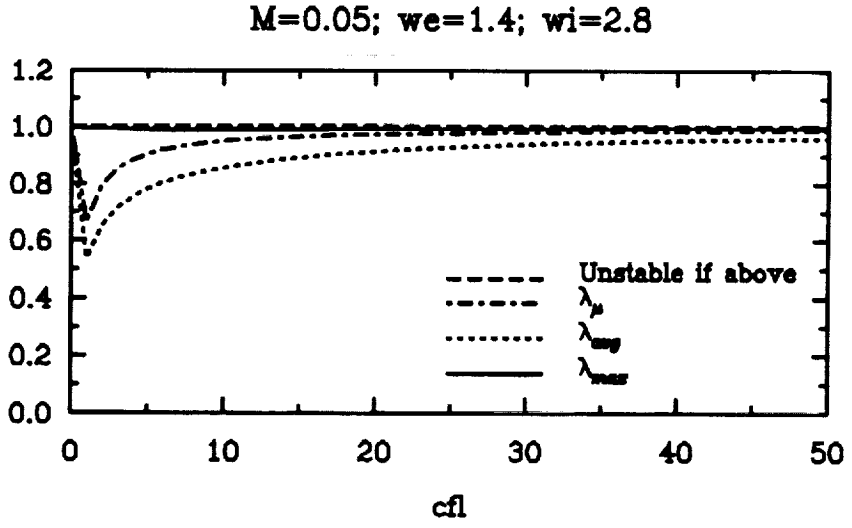
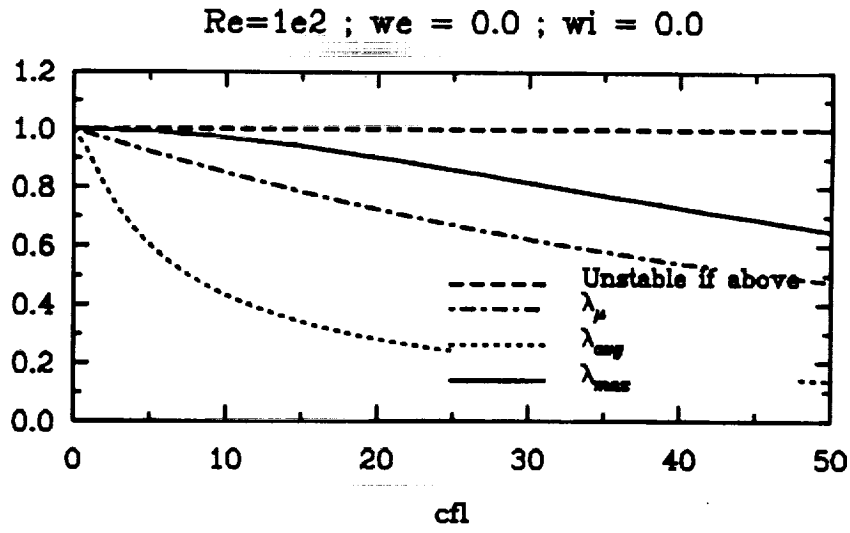


Fig. II.3.16 : Stability Analysis for 2–D Navier–Stokes Equations (Cent. Diff.) Re=1000000



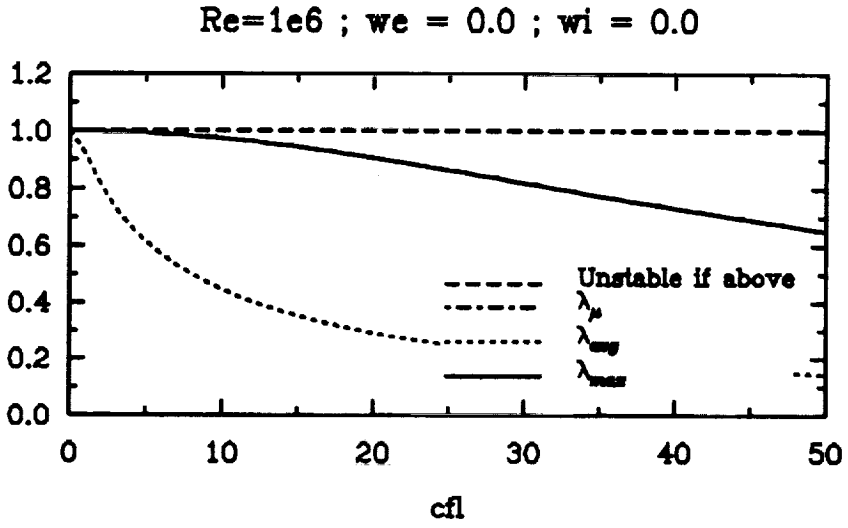
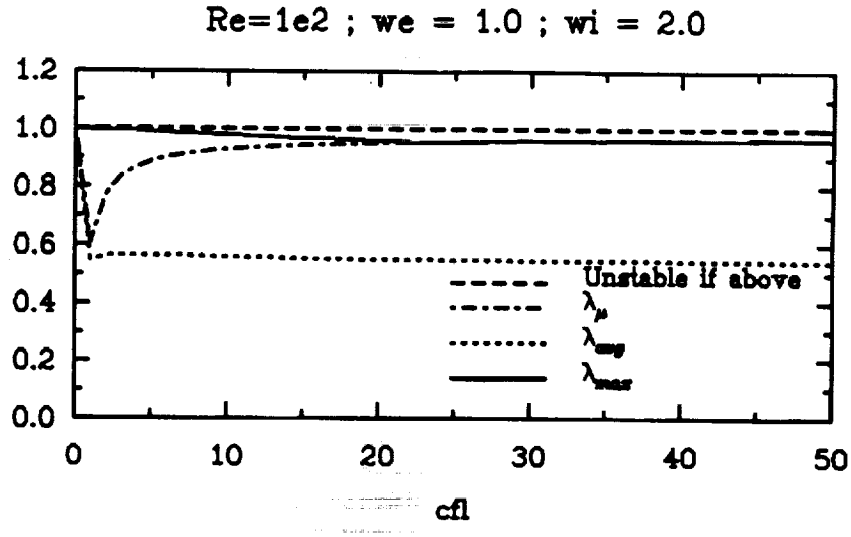


Fig. II.3.17 : Stability Analysis for 1–D Navier–Stokes Equations (Cent. Diff.)



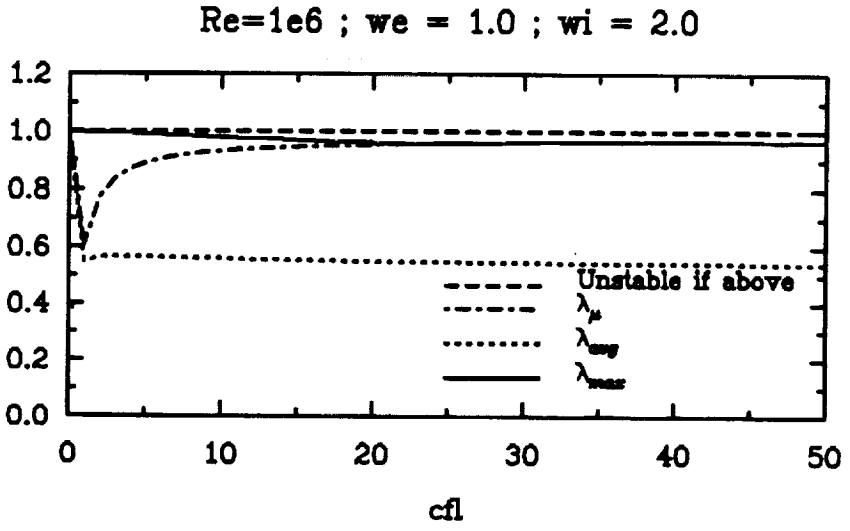
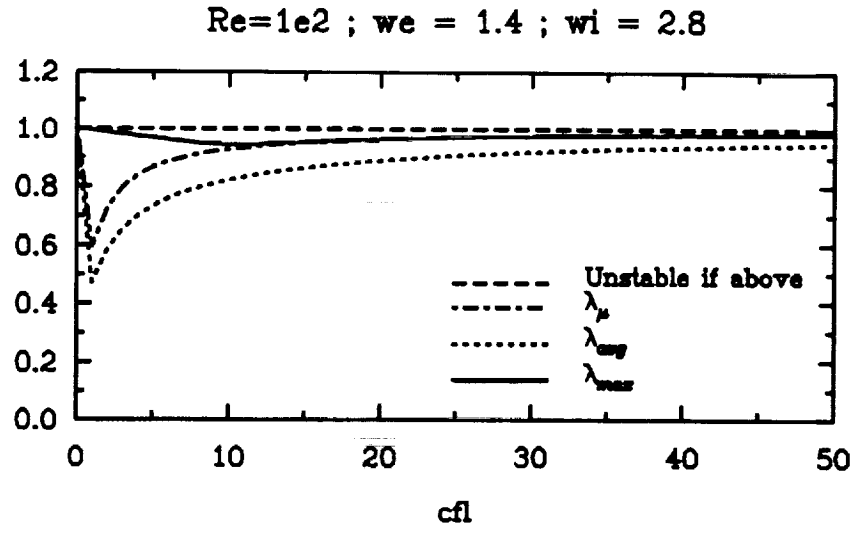


Fig. II.3.18 : Stability Analysis for 1–D Navier–Stokes Equations (Cent. Diff.)



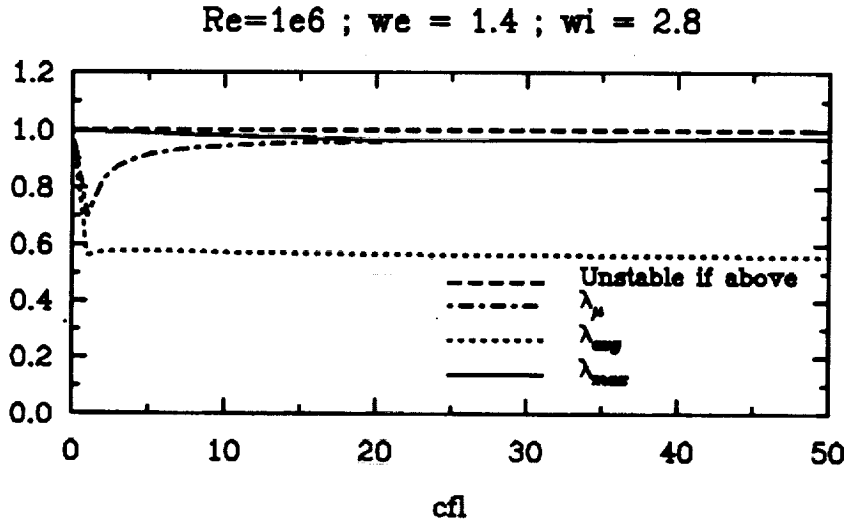


Fig. II.3.19 : Stability Analysis for 1–D Navier–Stokes Equations (Cent. Diff.)

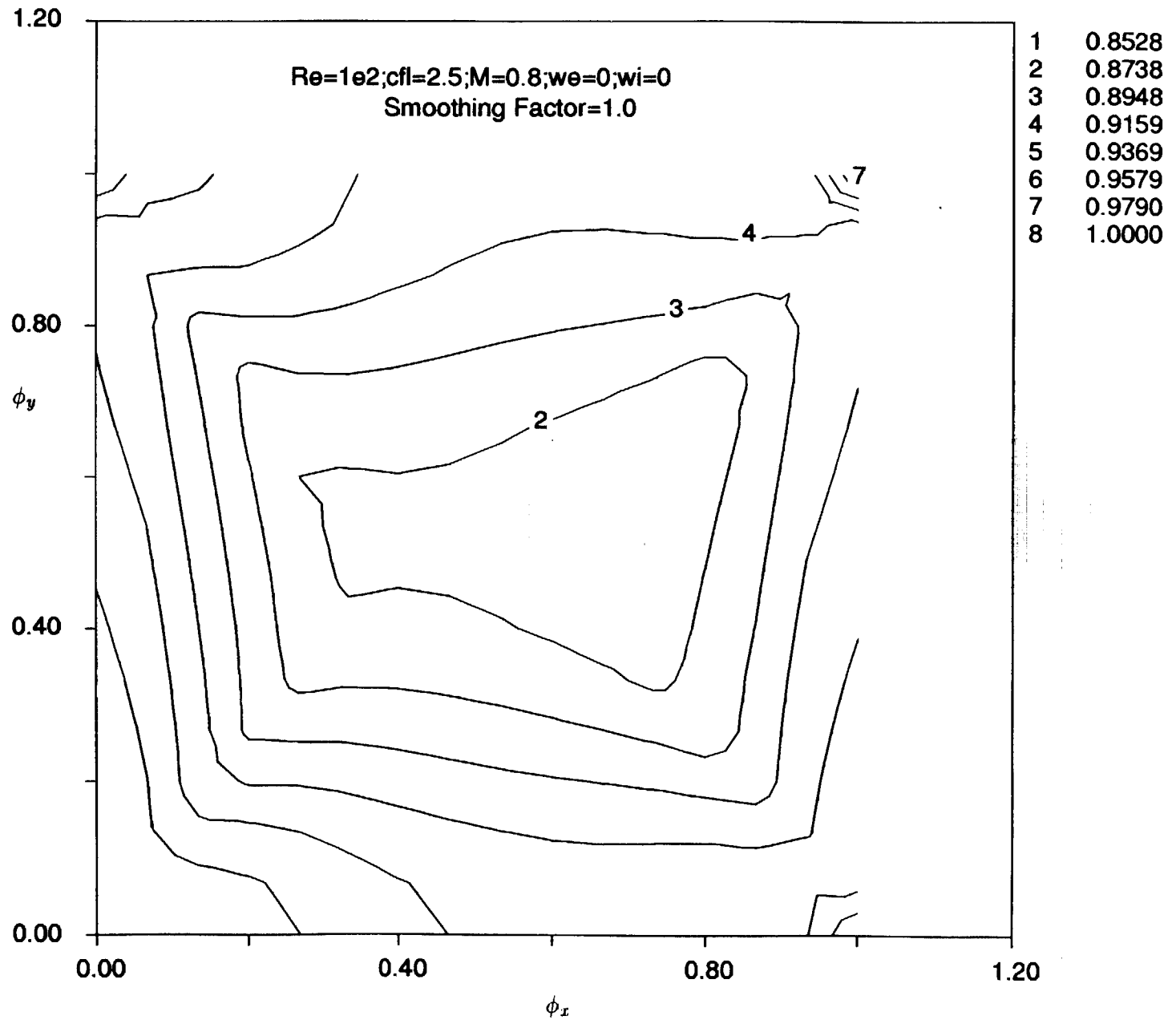


Fig. II.3.14 : Stability Analysis for 2-D Navier-Stokes Equations (Central Diff.)

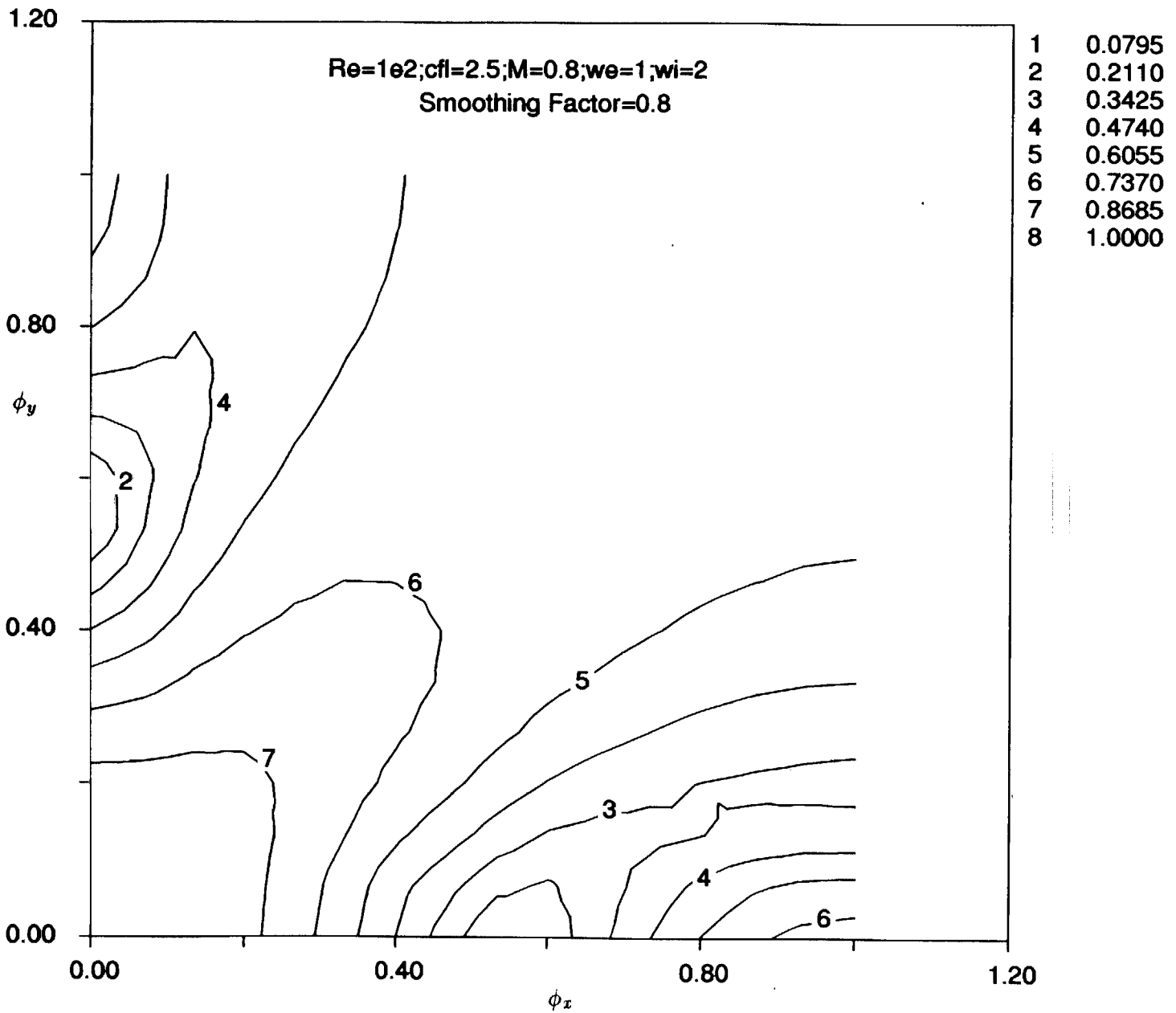


Fig. II.3.15 : Stability Analysis for 2-D Navier-Stokes Equations (Central Diff.)

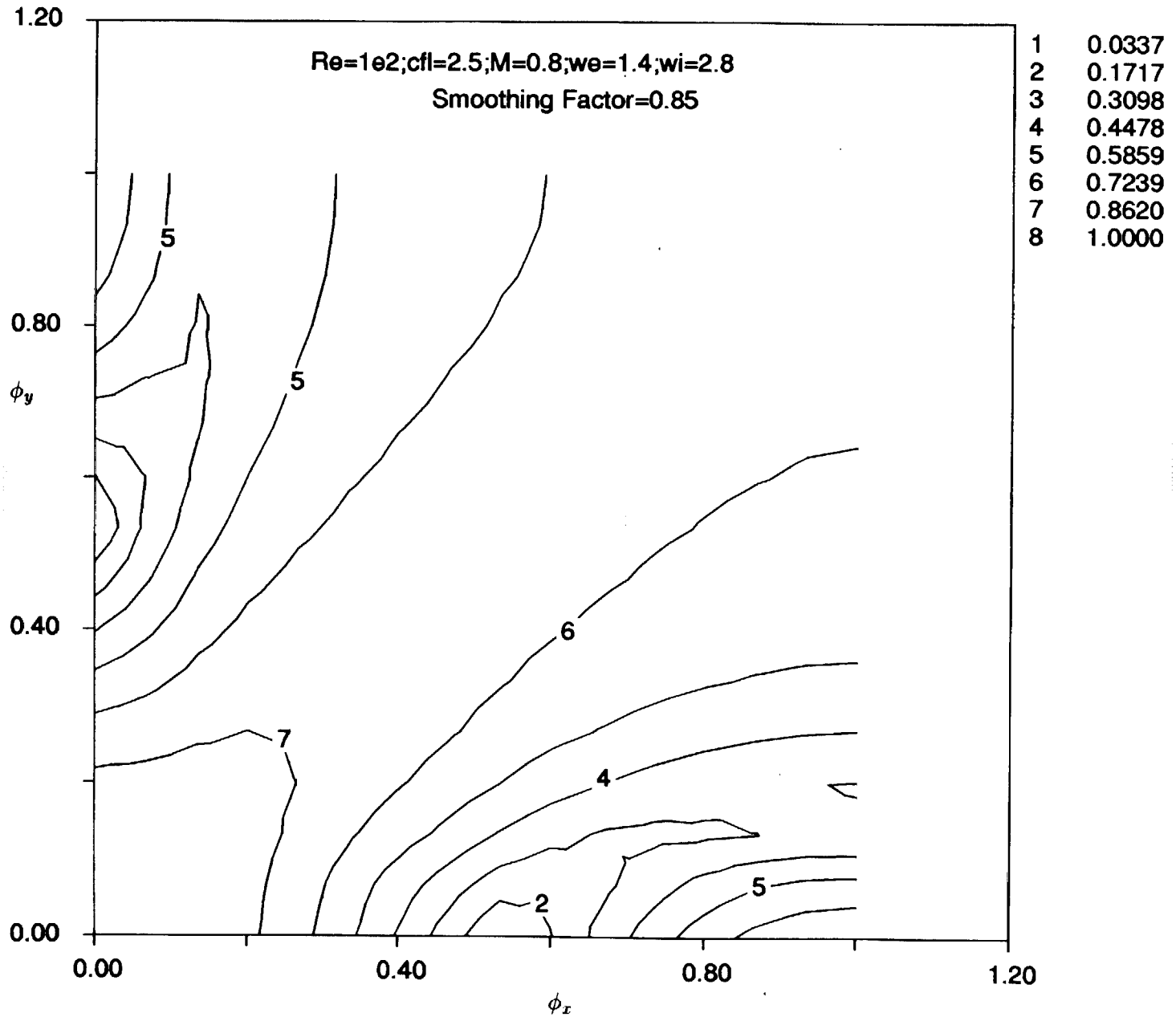


Fig. II.3.16 : Stability Analysis for 2-D Navier-Stokes Equations (Central Diff.)

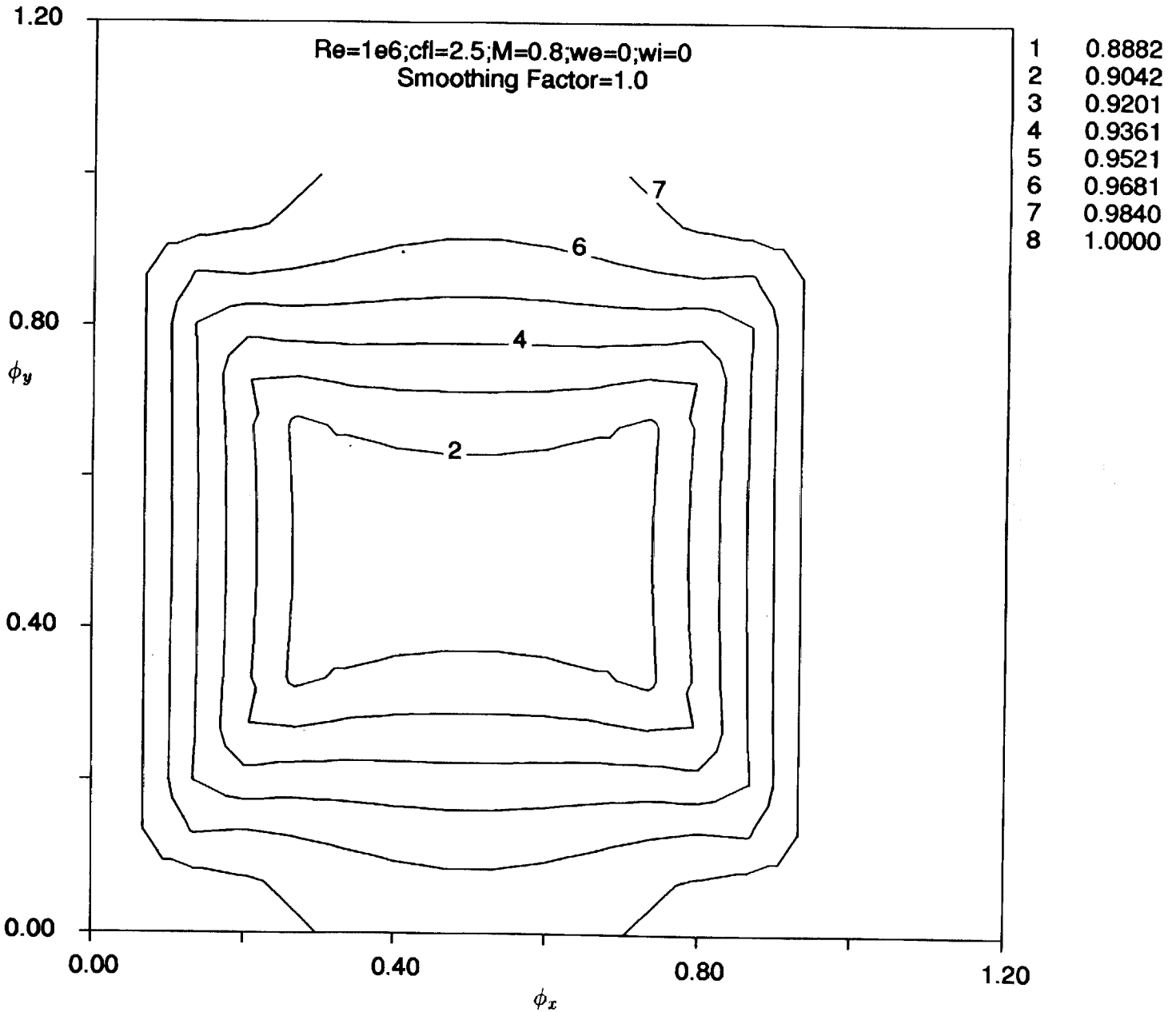


Fig. II.3.17 : Stability Analysis for 2-D Navier-Stokes Equations (Central Diff.)

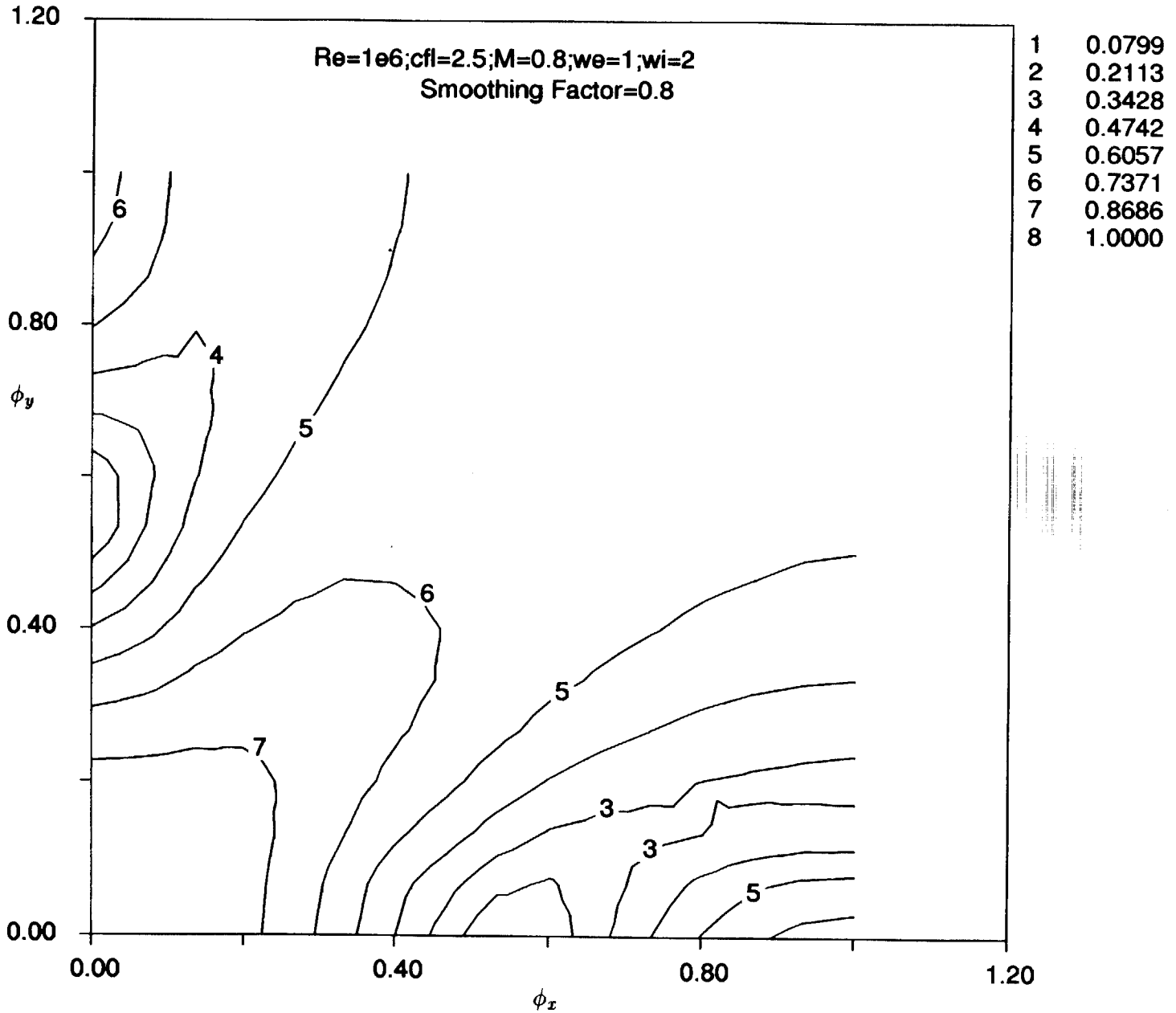


Fig. II.3.18 : Stability Analysis for 2-D Navier-Stokes Equations (Central Diff.)

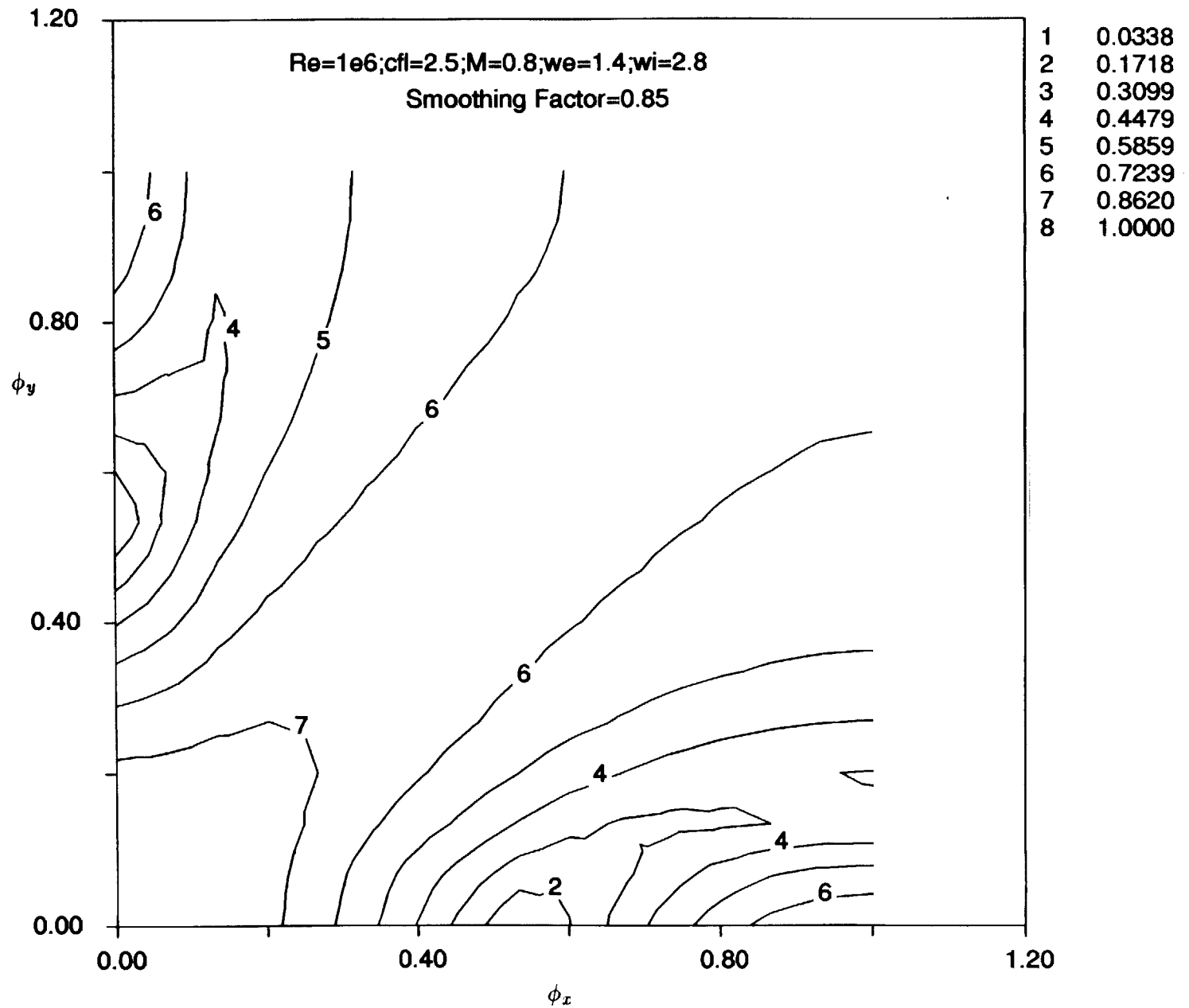


Fig. II.3.19 : Stability Analysis for 2-D Navier-Stokes Equations (Central Diff.)

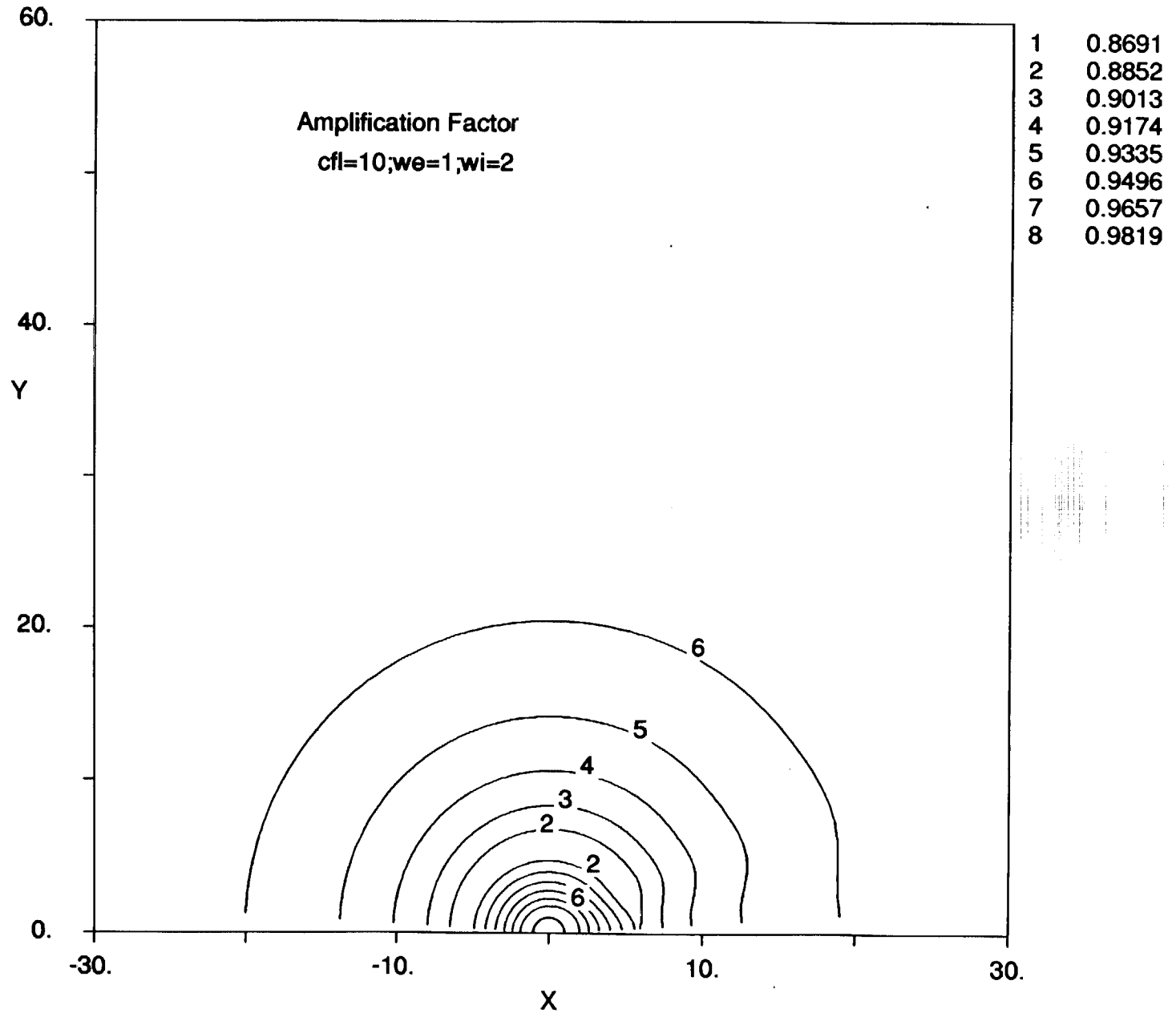


Fig. II.4.1 : Viscous Flow Past a Circular Cylinder

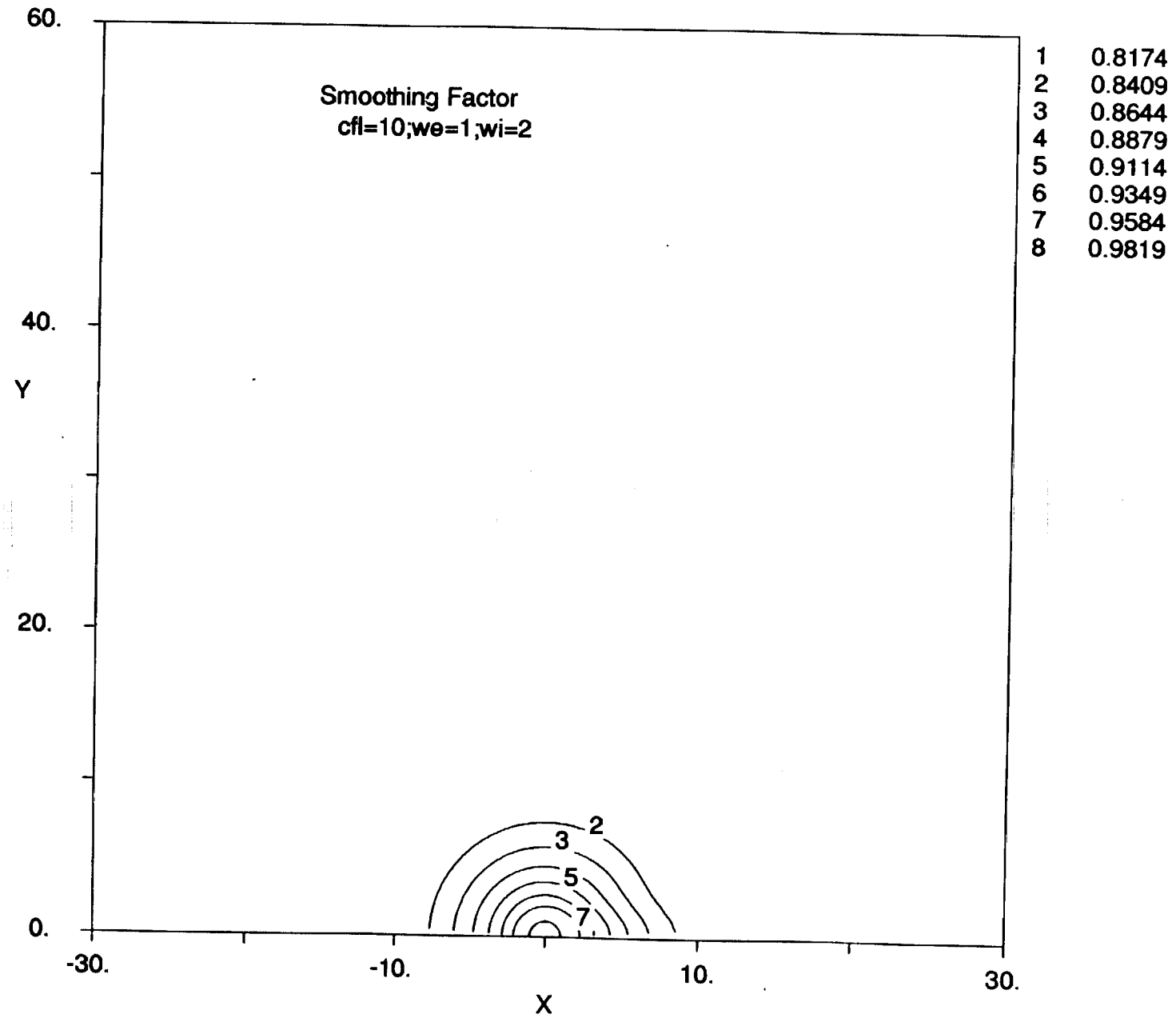


Fig. II.4.2 : Viscous Flow Past a Circular Cylinder

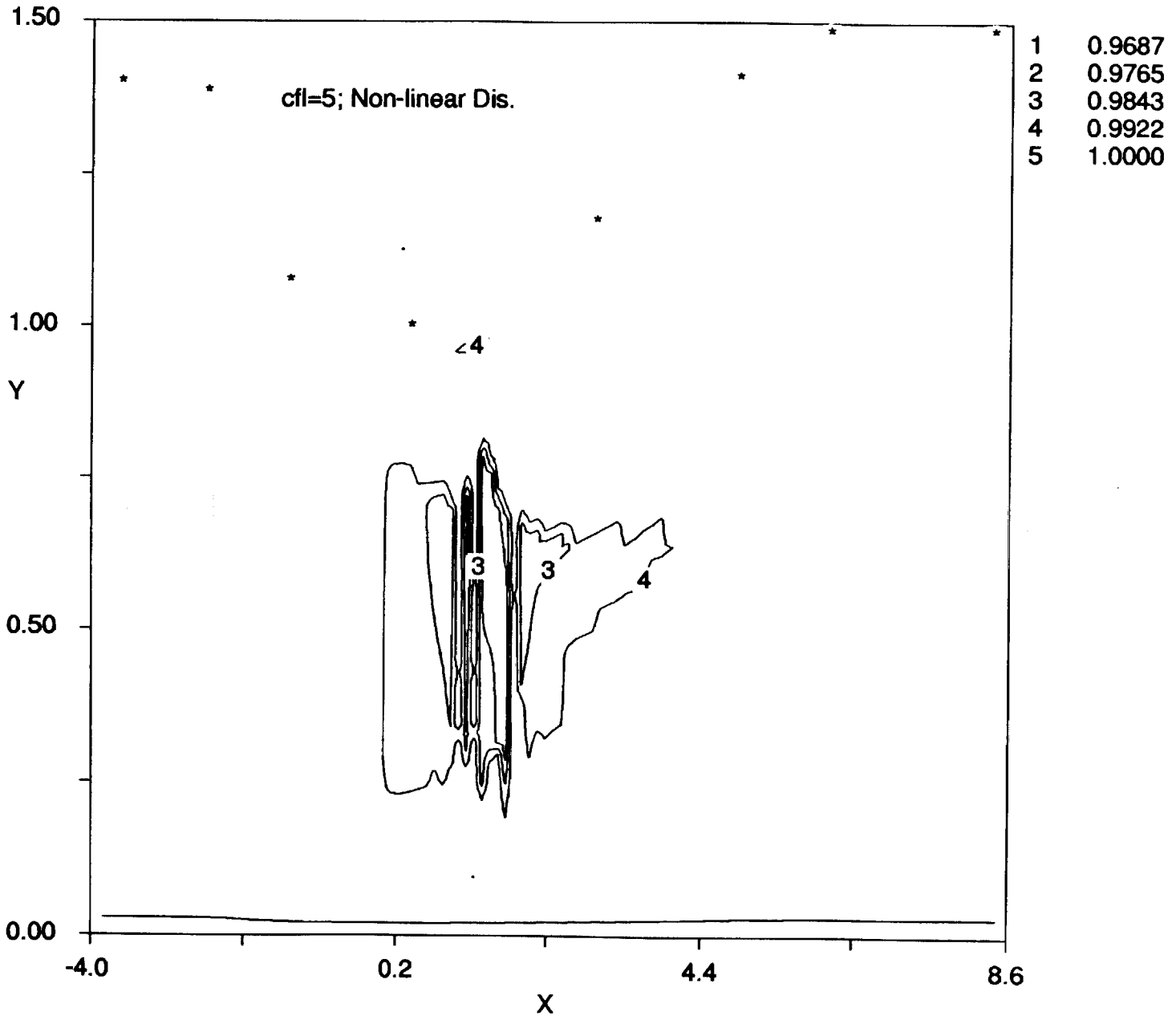


Fig. II.4.3 : Sajben Transonic Diffuser Flow

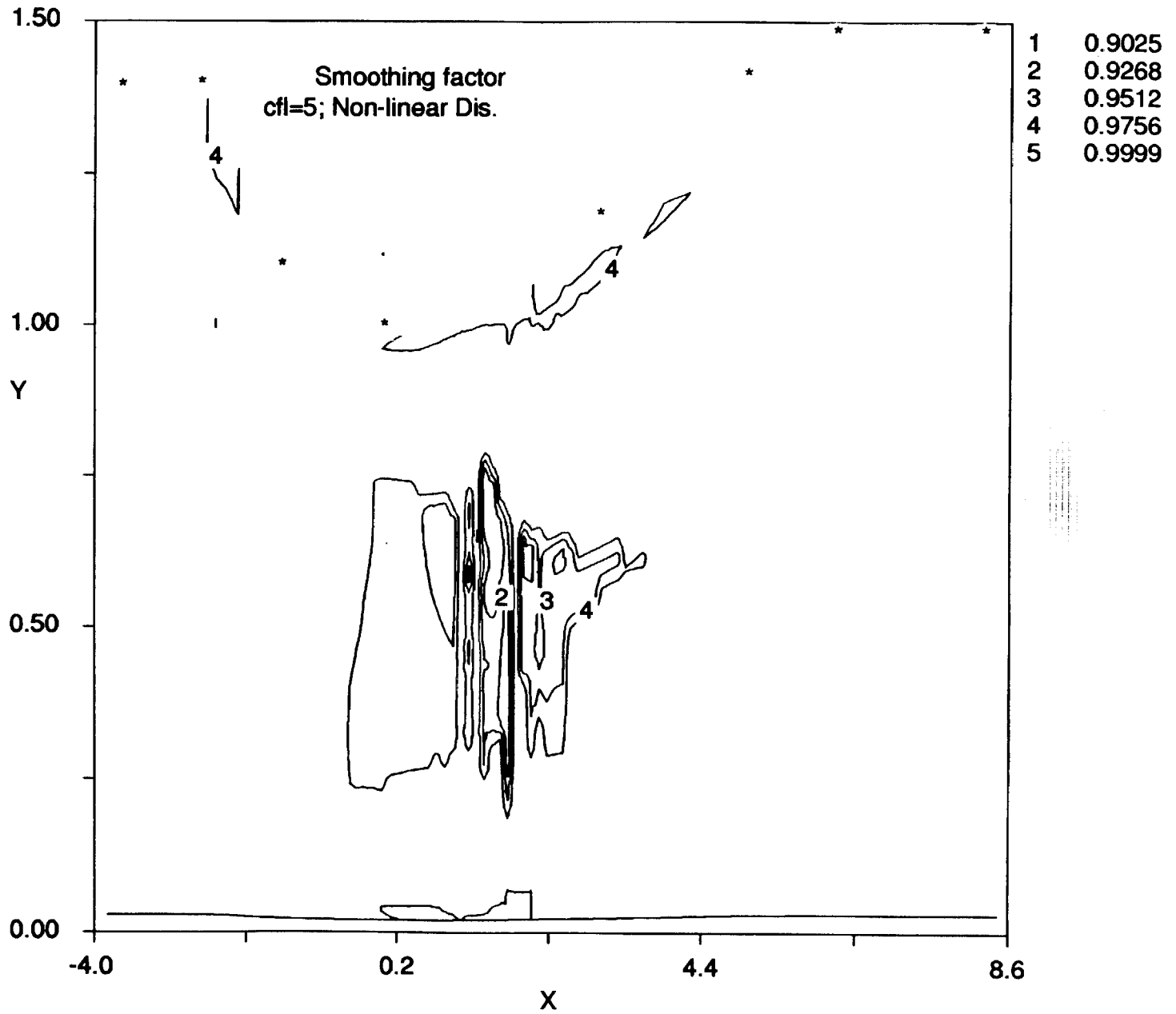


Fig. II.4.4 : Sajben Transonic Diffuser Flow

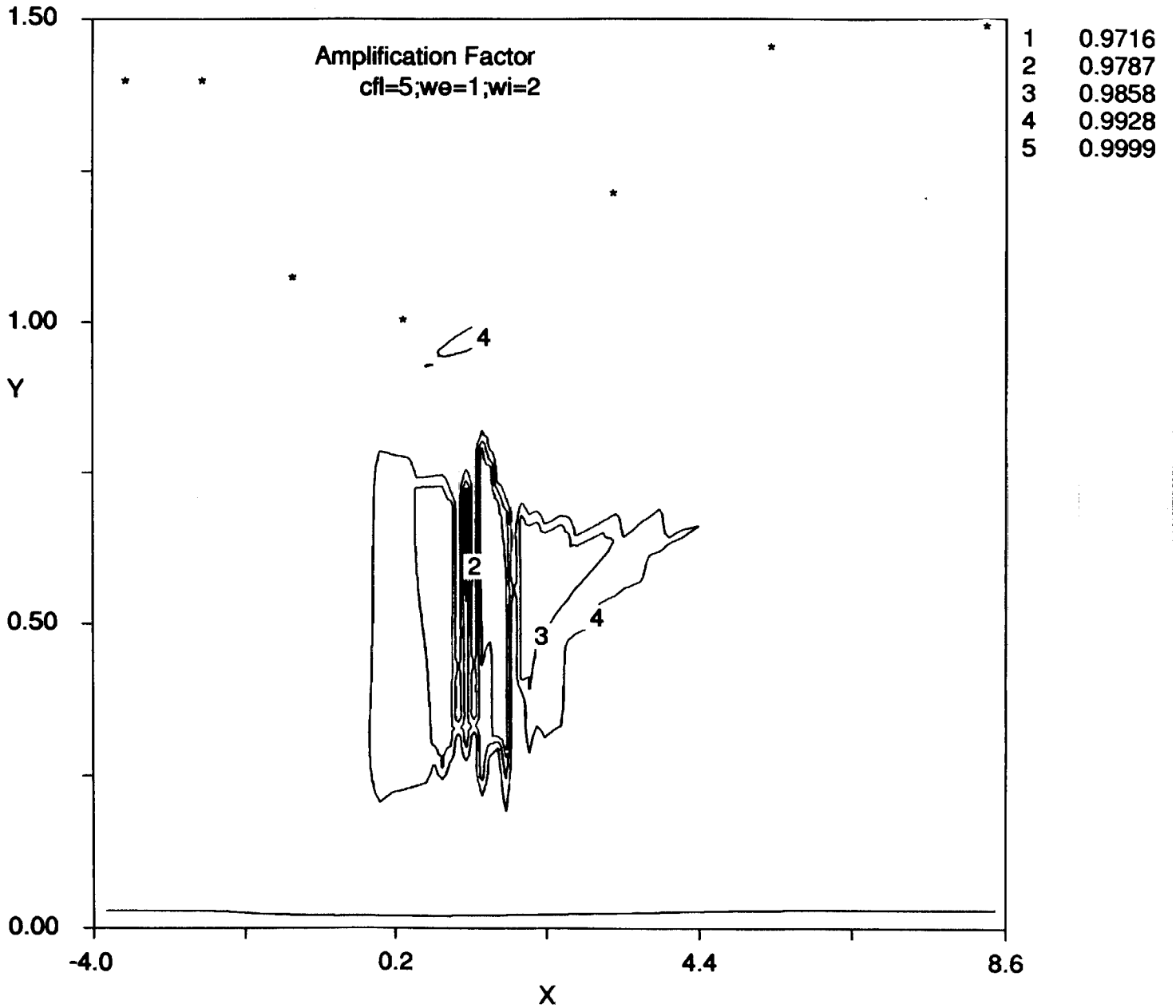


Fig. II.4.5 : Sajben Transonic Diffuser Flow

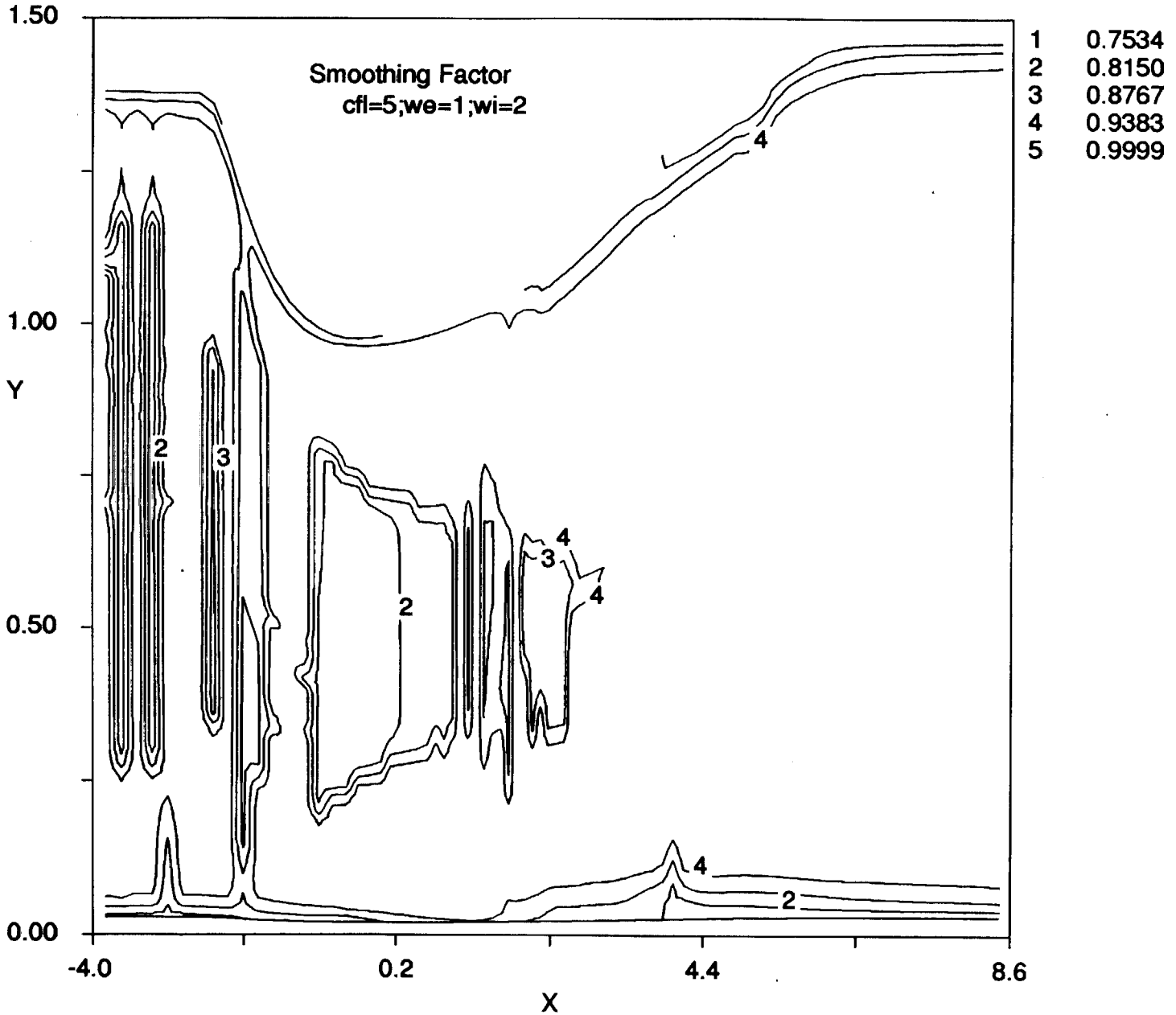


Fig. II.4.6 : Sajben Transonic Diffuser Flow

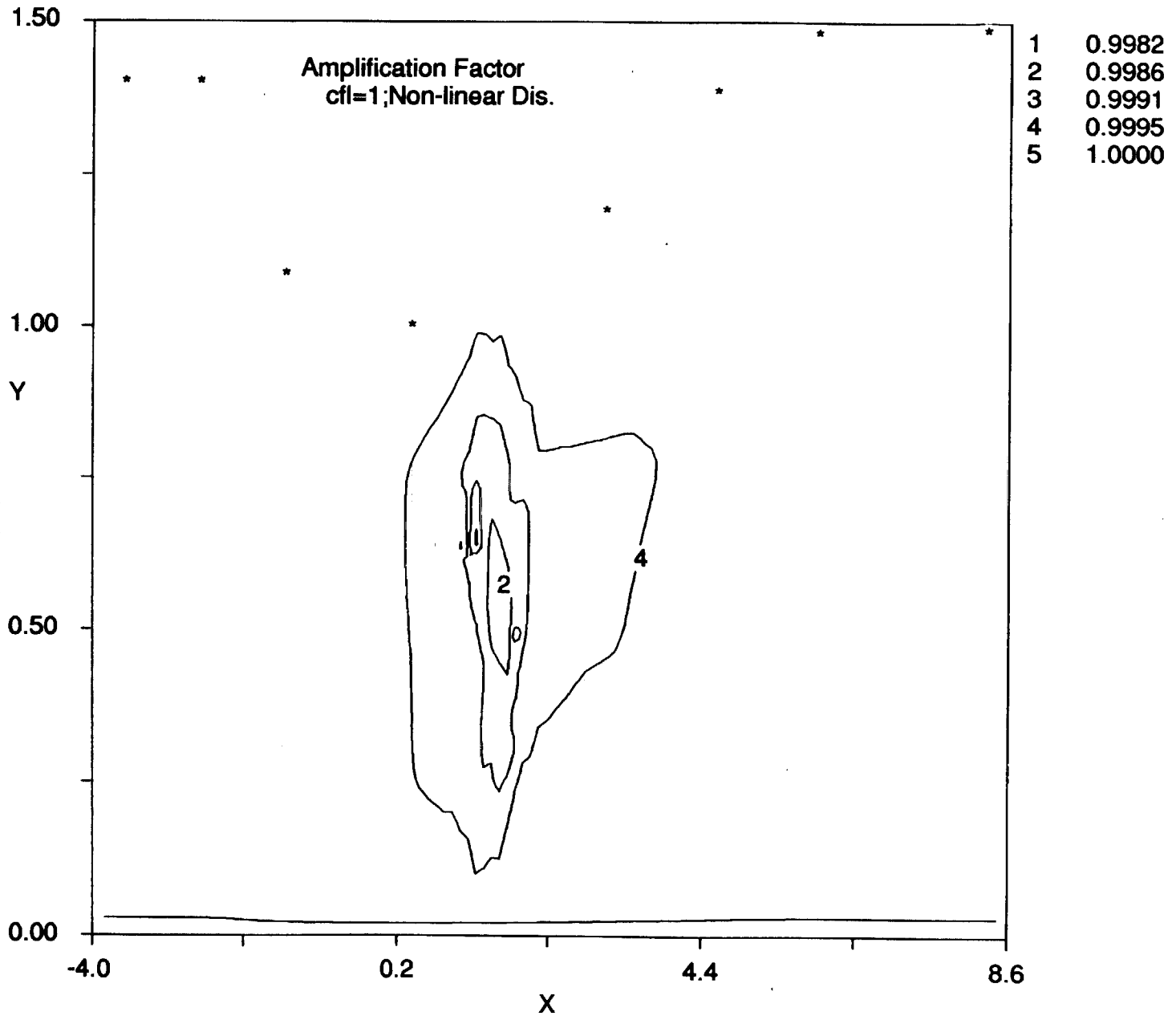


Fig. II.4.7 : Sajben Transonic Diffuser Flow

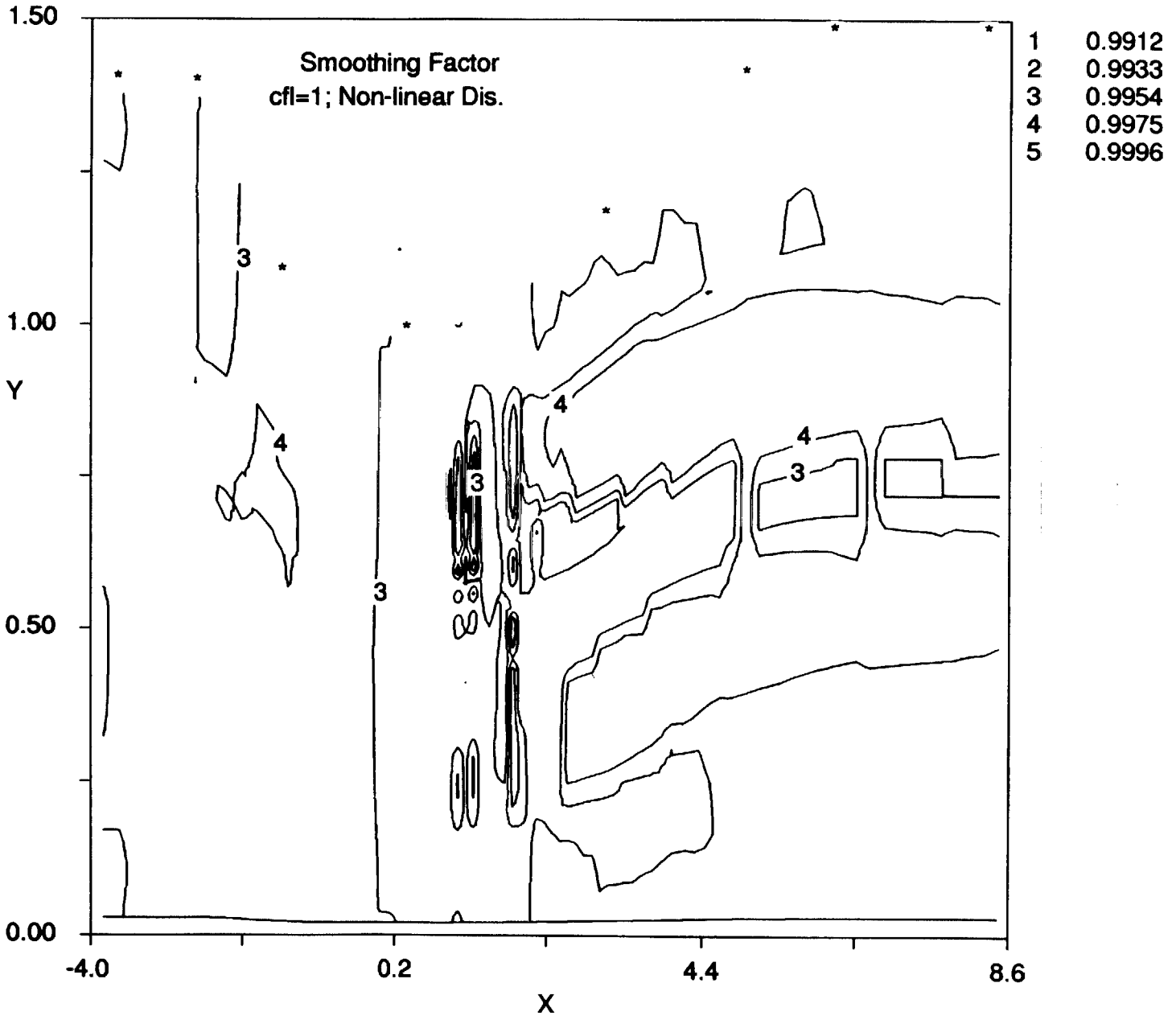


Fig. II.4.8 : Sajben Transonic Diffuser Flow

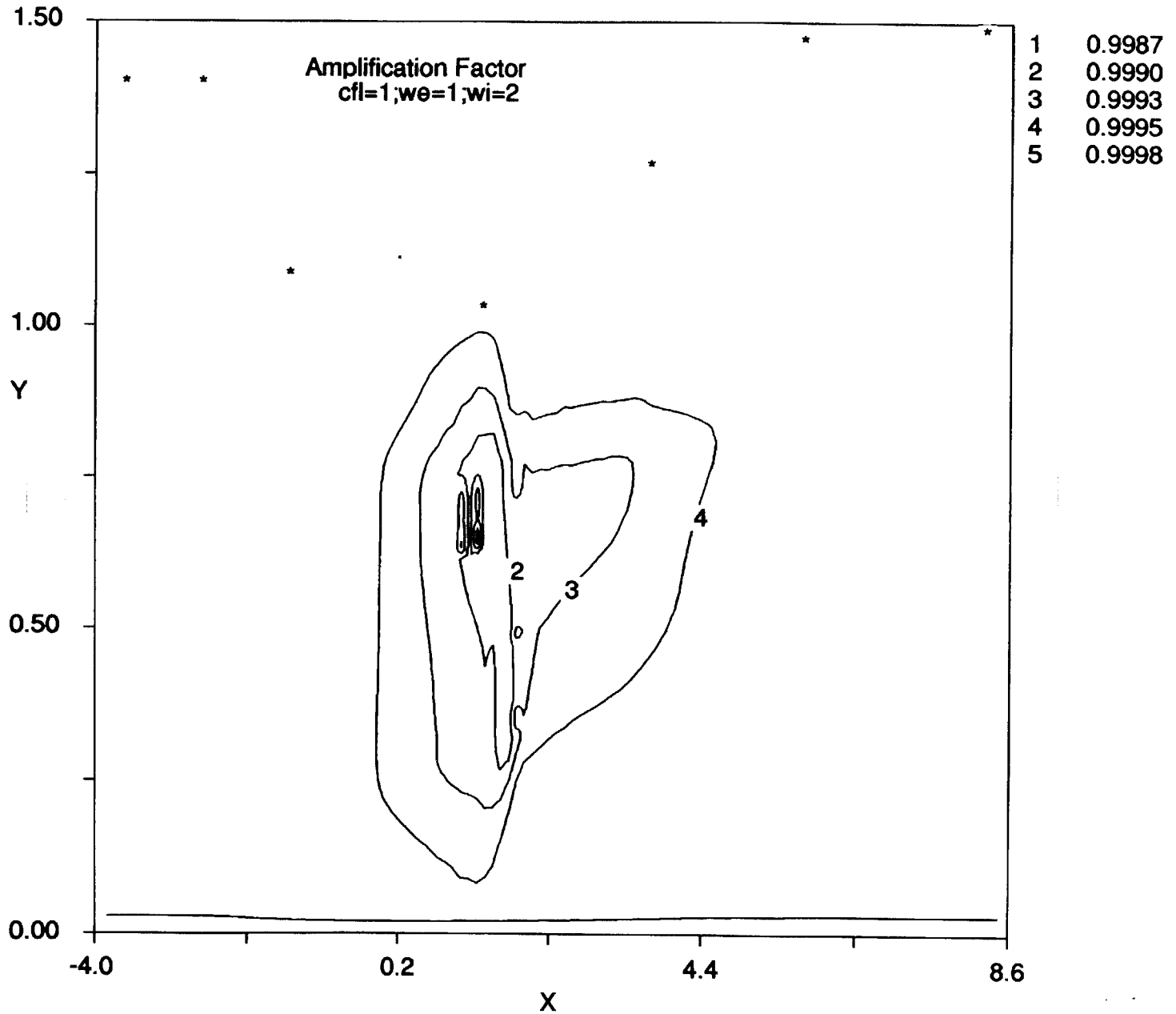


Fig. II.4.9 : Sajben Transonic Diffuser Flow

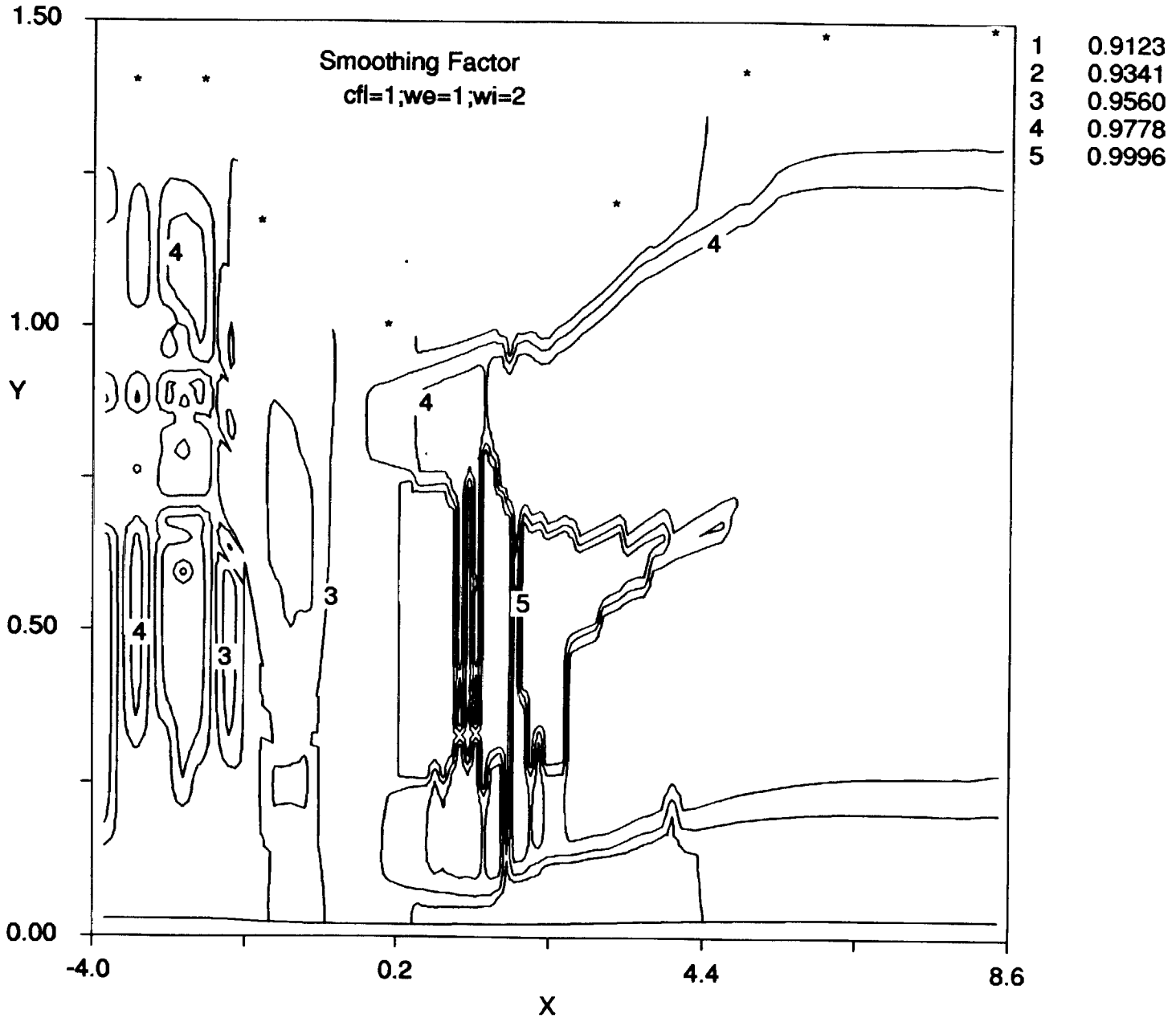


Fig. II.4.10 : Sajben Transonic Diffuser Flow

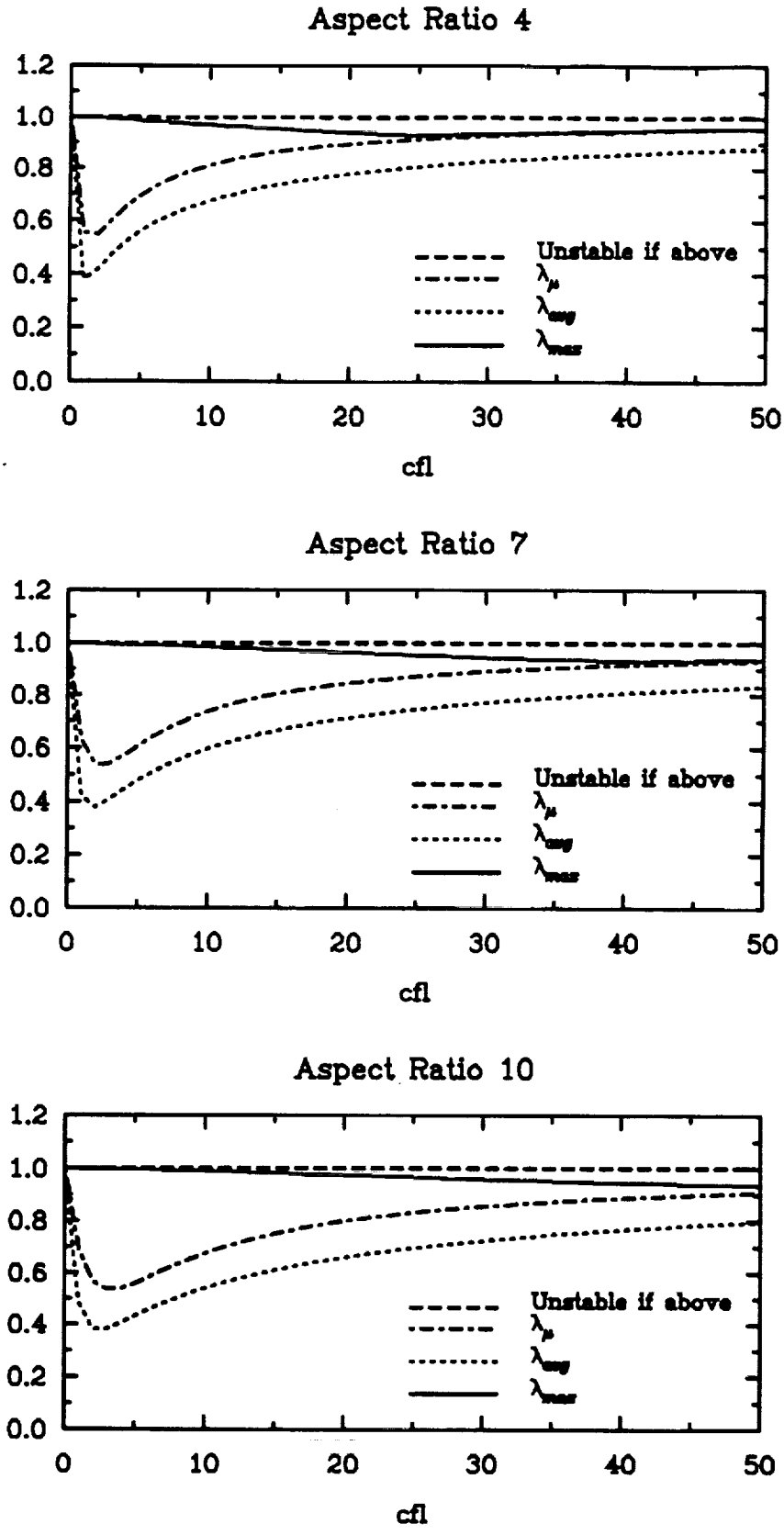


Fig. II.4.11 : Stability Analysis for 2-D Navier-Stokes Equations (Cent. Diff.) (Re=1000000, M=0.8, we=1, wi=2)

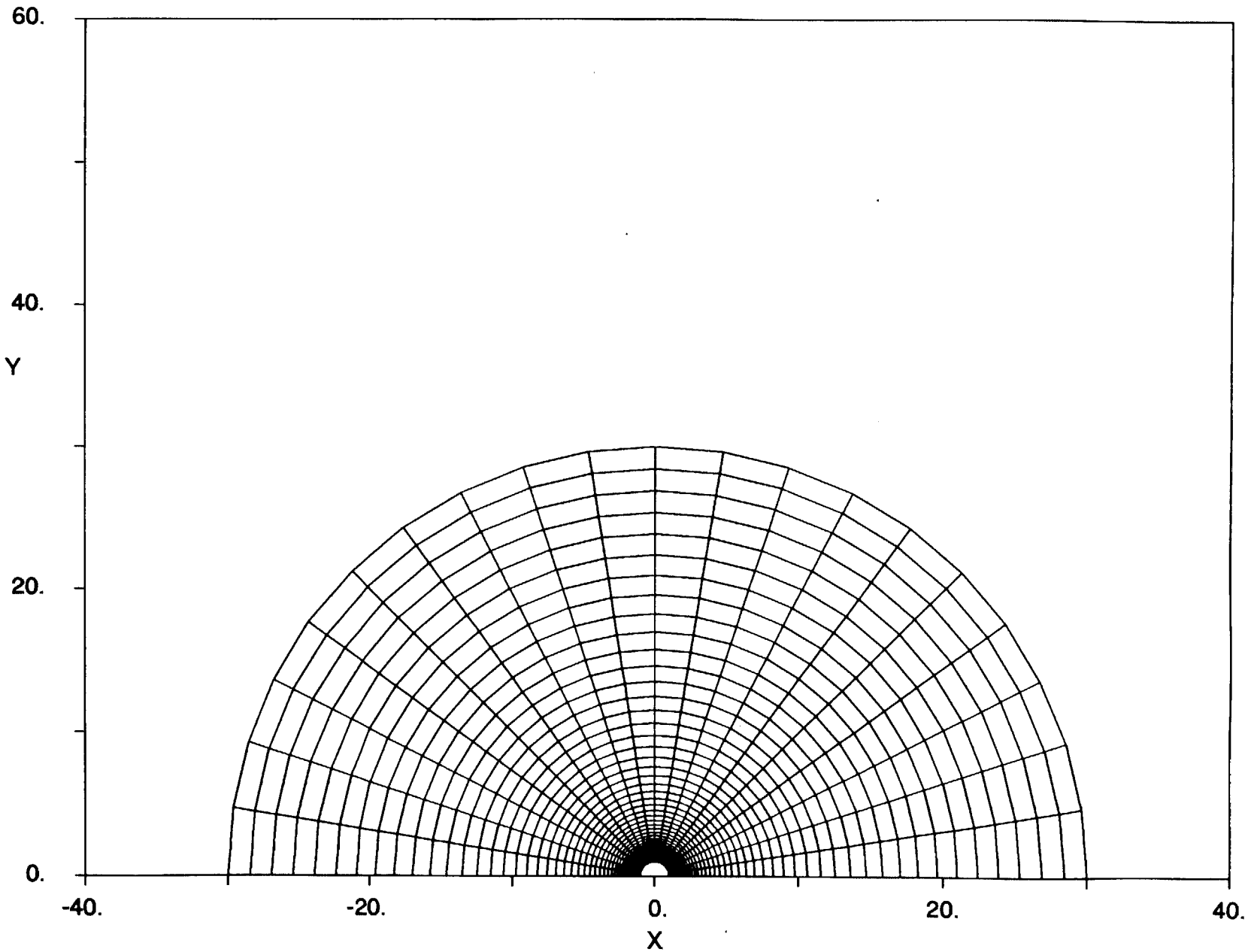


Fig. III.1 : Base-Case Grid for Euler Flow Past a Circular Cylinder

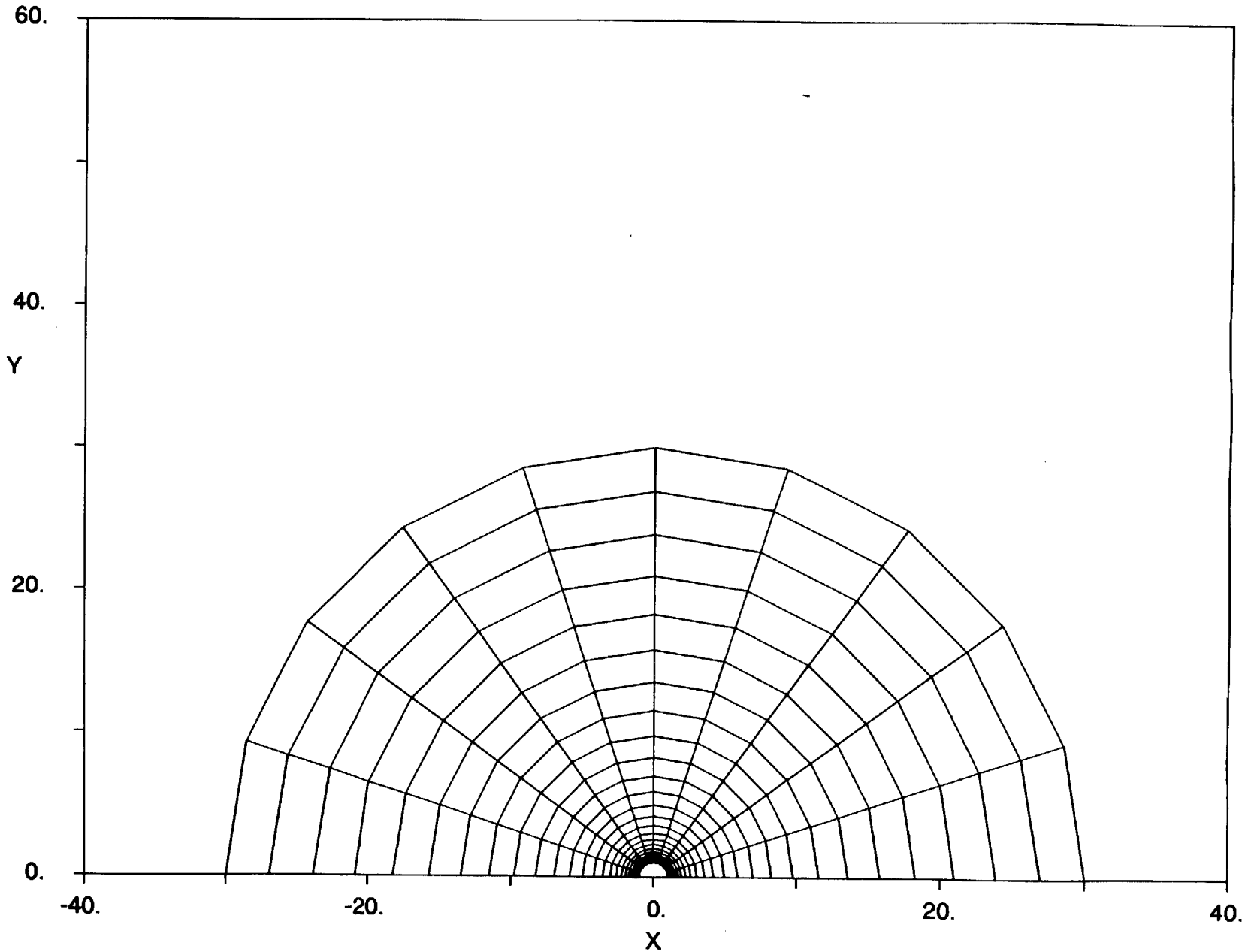


Fig. III.2 : Twice as Coarse Grid for Euler Flow Past a Circular Cylinder

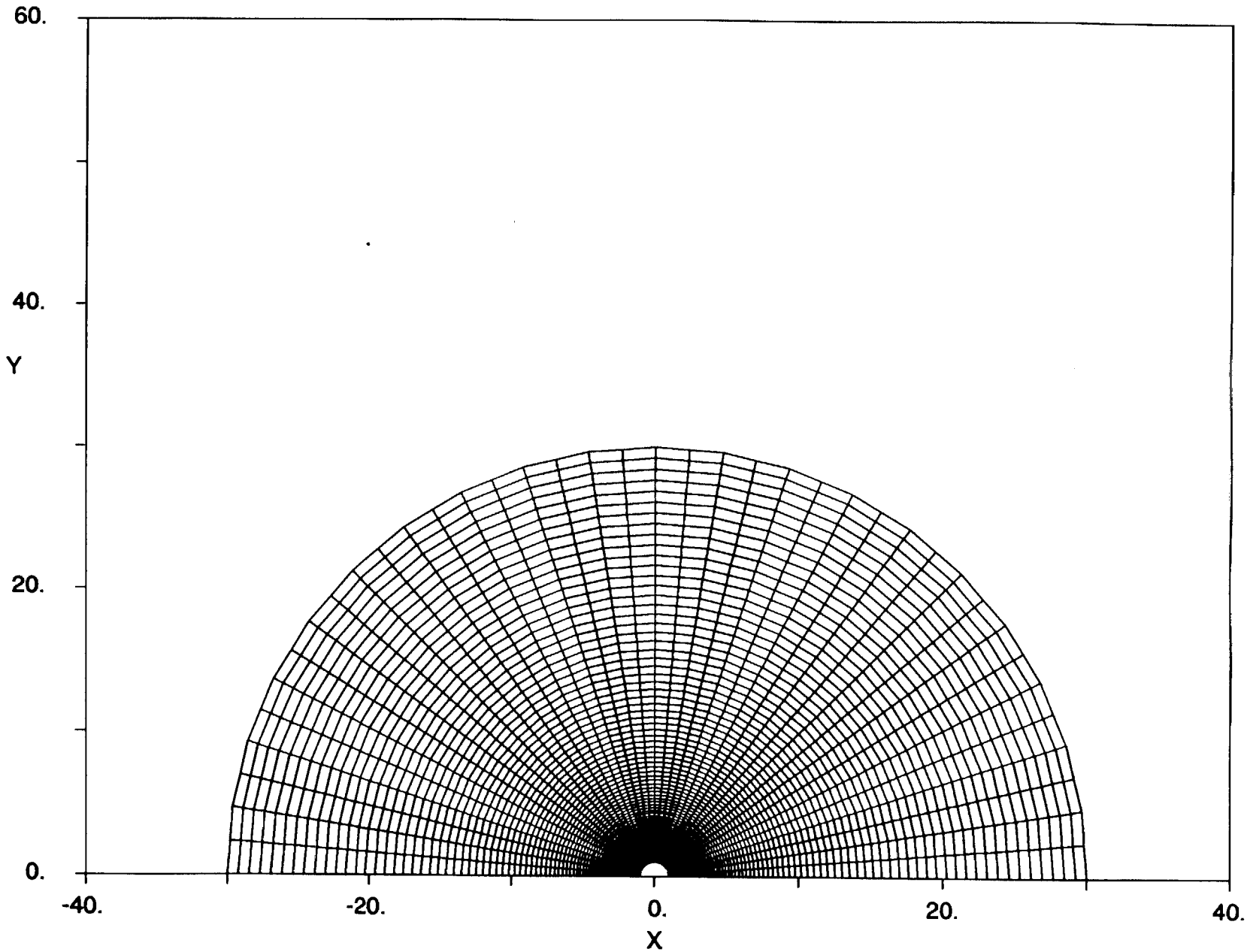


Fig. III.3 : Twice as Fine Grid for Euler Flow Past a Circular Cylinder

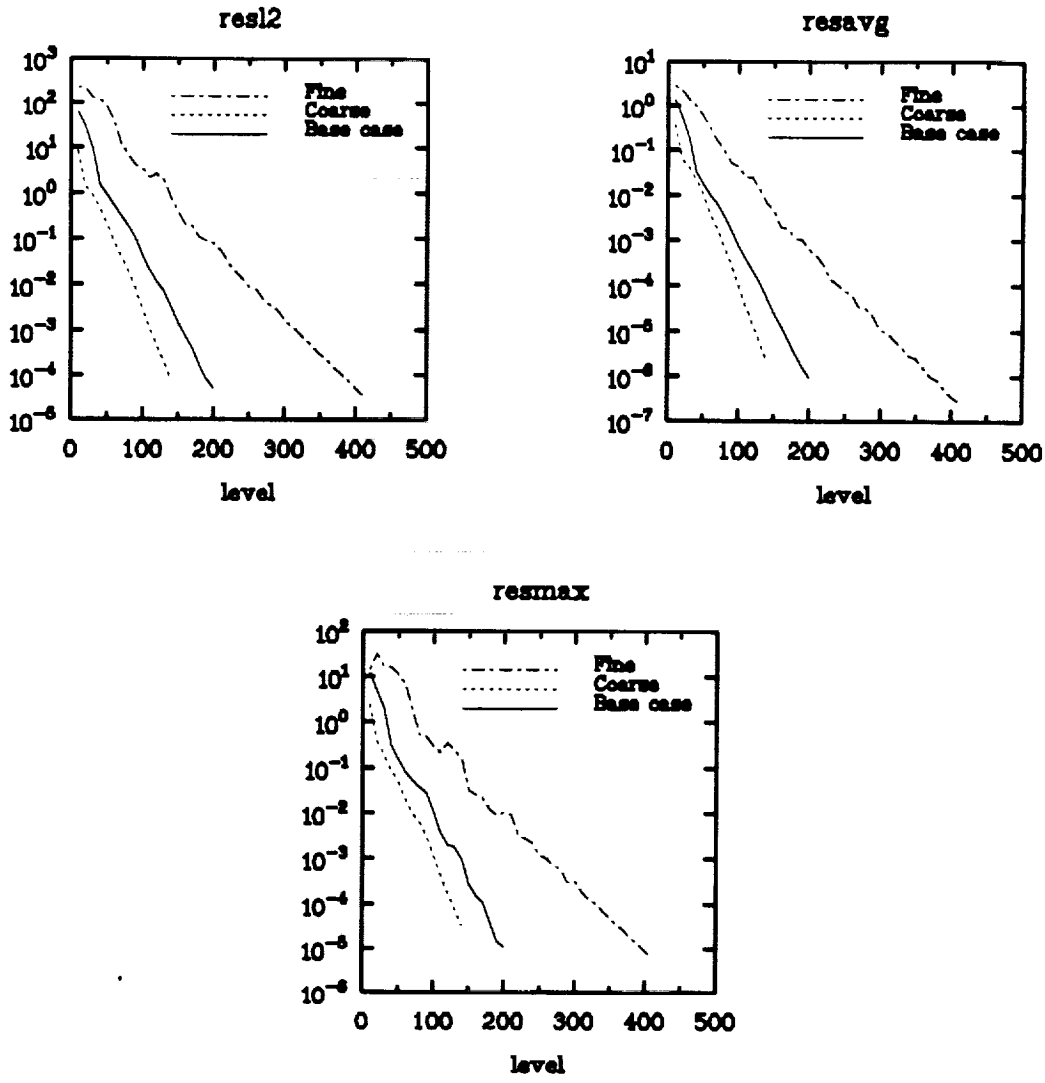


Fig. III.4 : Euler flow past a circular cylinder
Convergence history for continuity equation
(Artificial Viscosity)
Base case grid 21*51

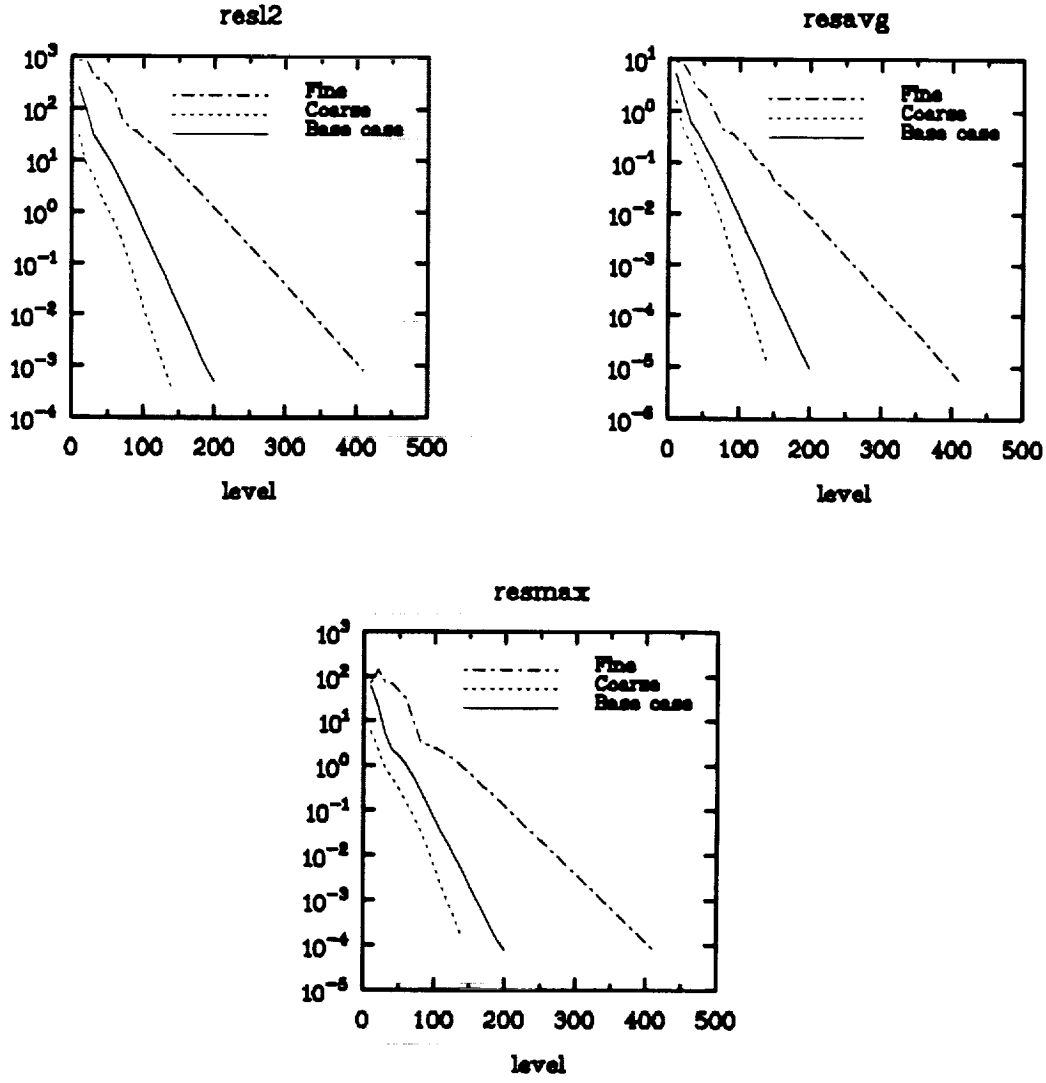


Fig. III.5 : Euler flow past a circular cylinder
Convergence history for x-momentum equation
(Artificial Viscosity)
Base case grid 21*51

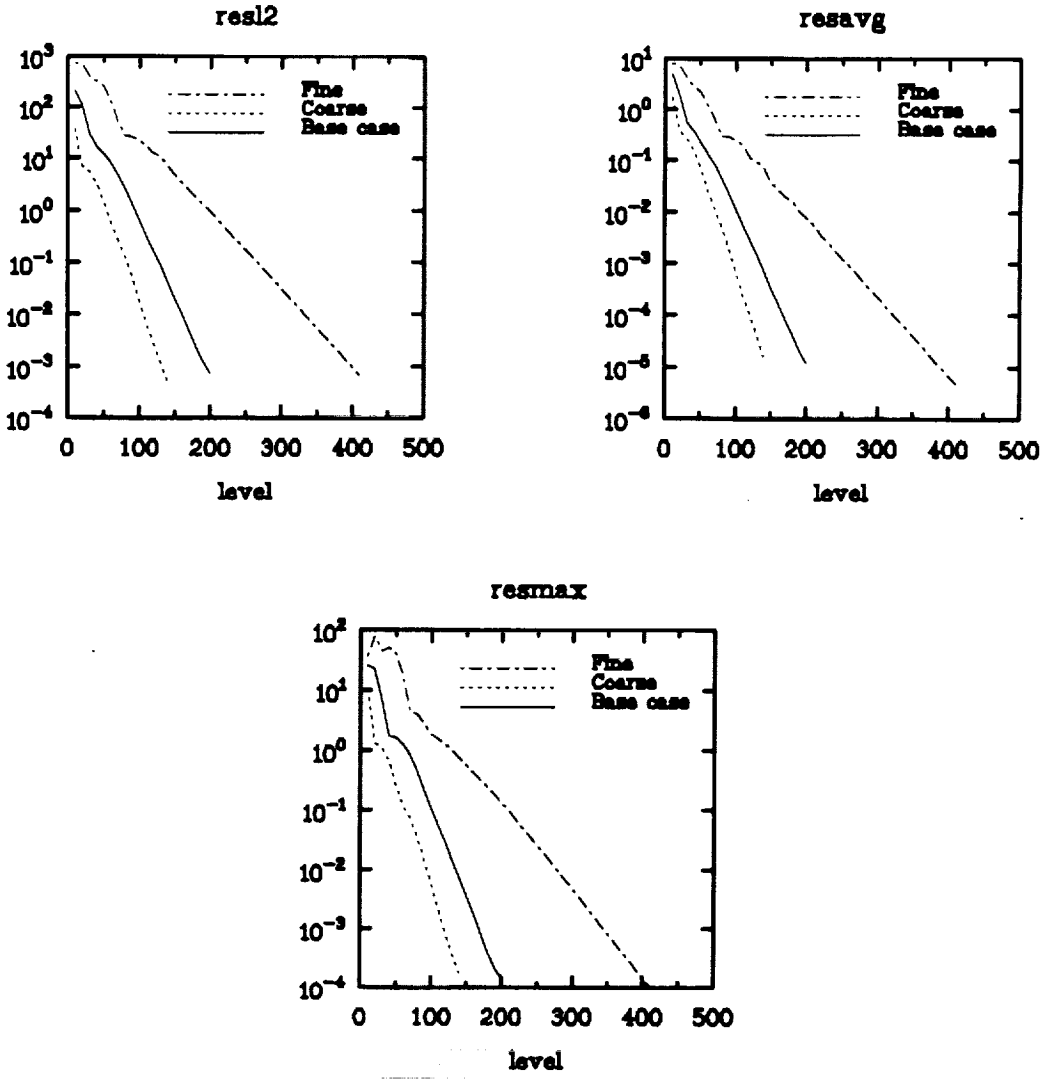


Fig. III.6 : Euler flow past a circular cylinder
Convergence history for y-momentum equation
(Artificial Viscosity)
Base case grid 21*51

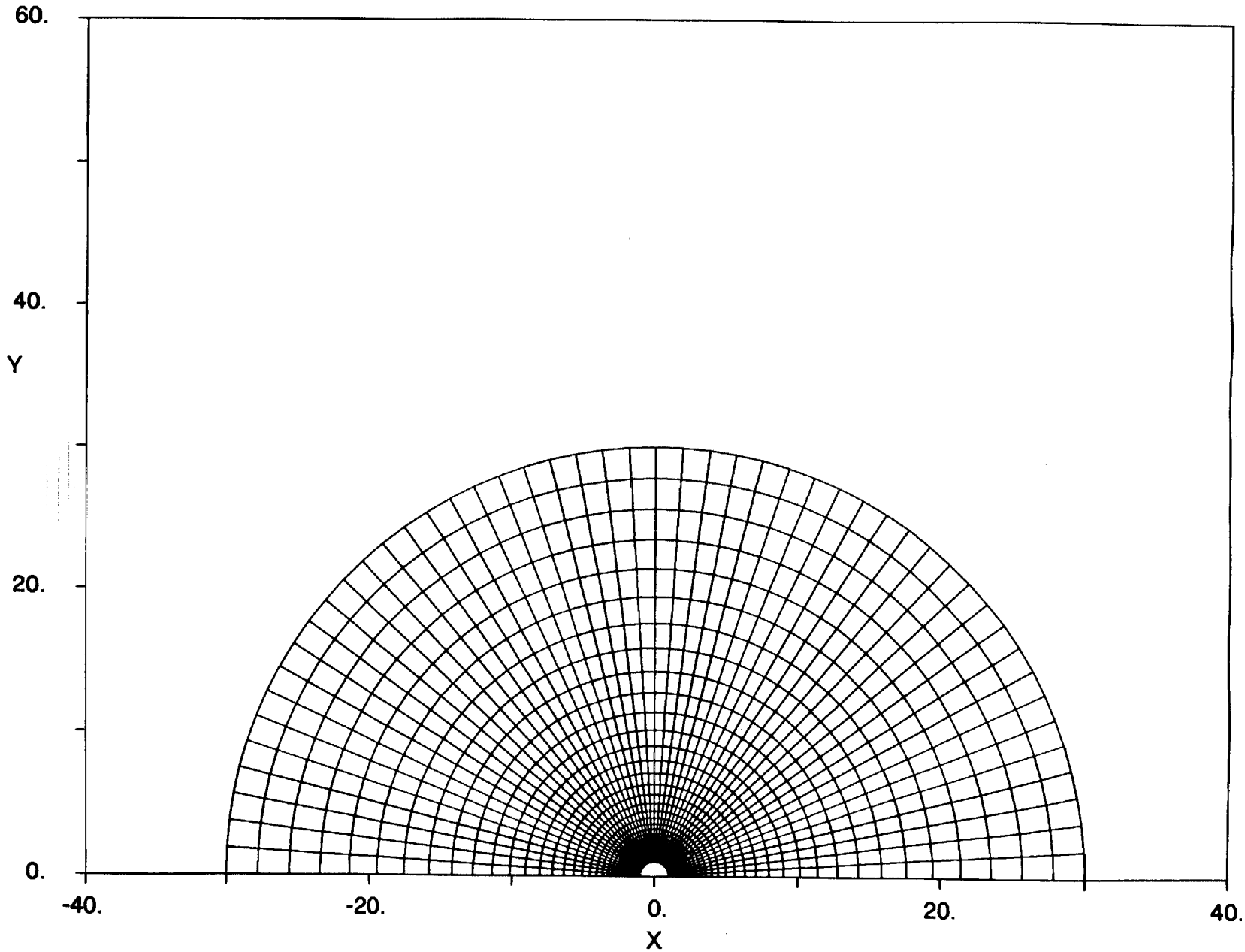


Fig. III.7 : Base-Case Grid for Viscous Flow Past a Circular Cylinder

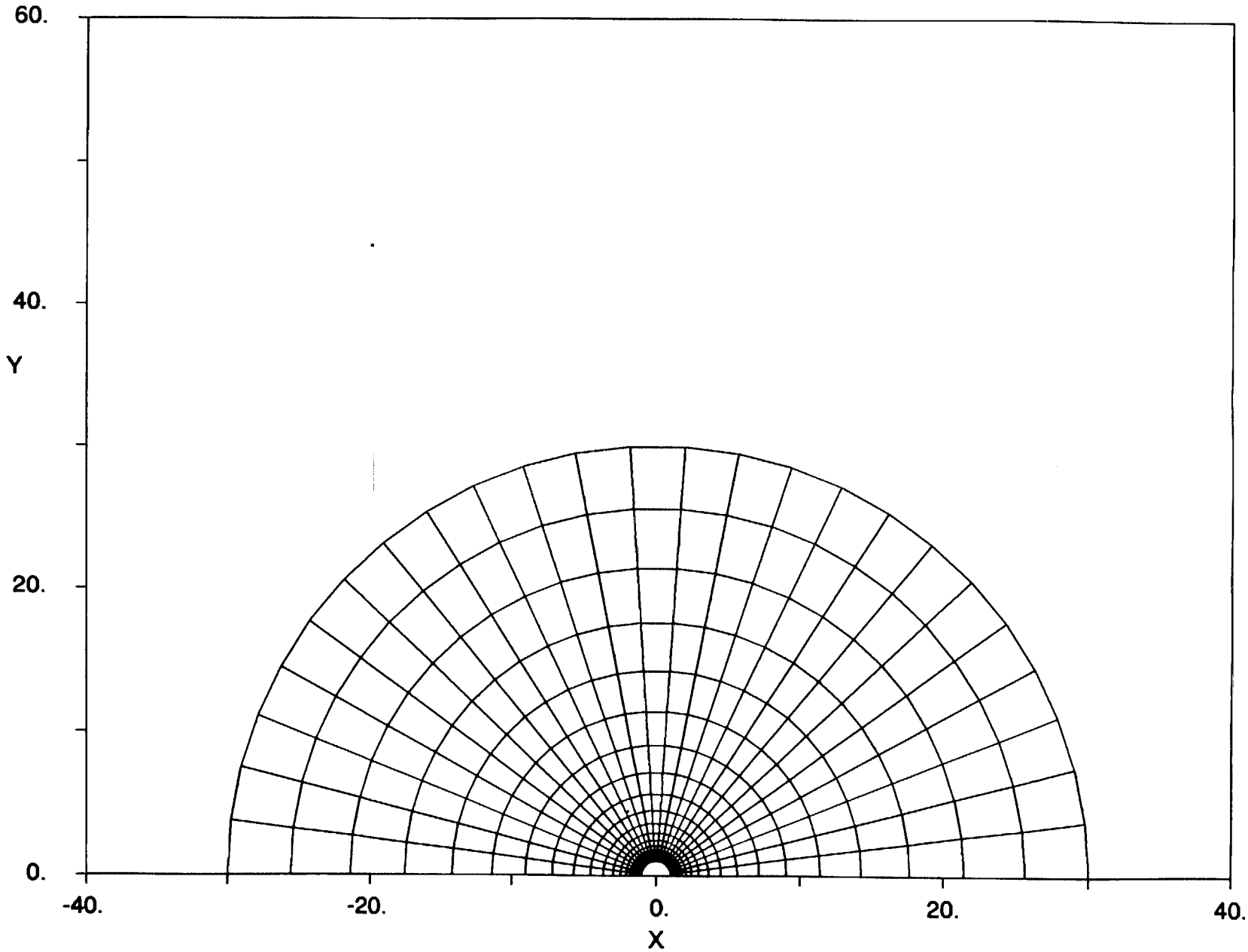


Fig. III.8 : Twice as Coarse Grid for Viscous Flow Past a Circular Cylinder

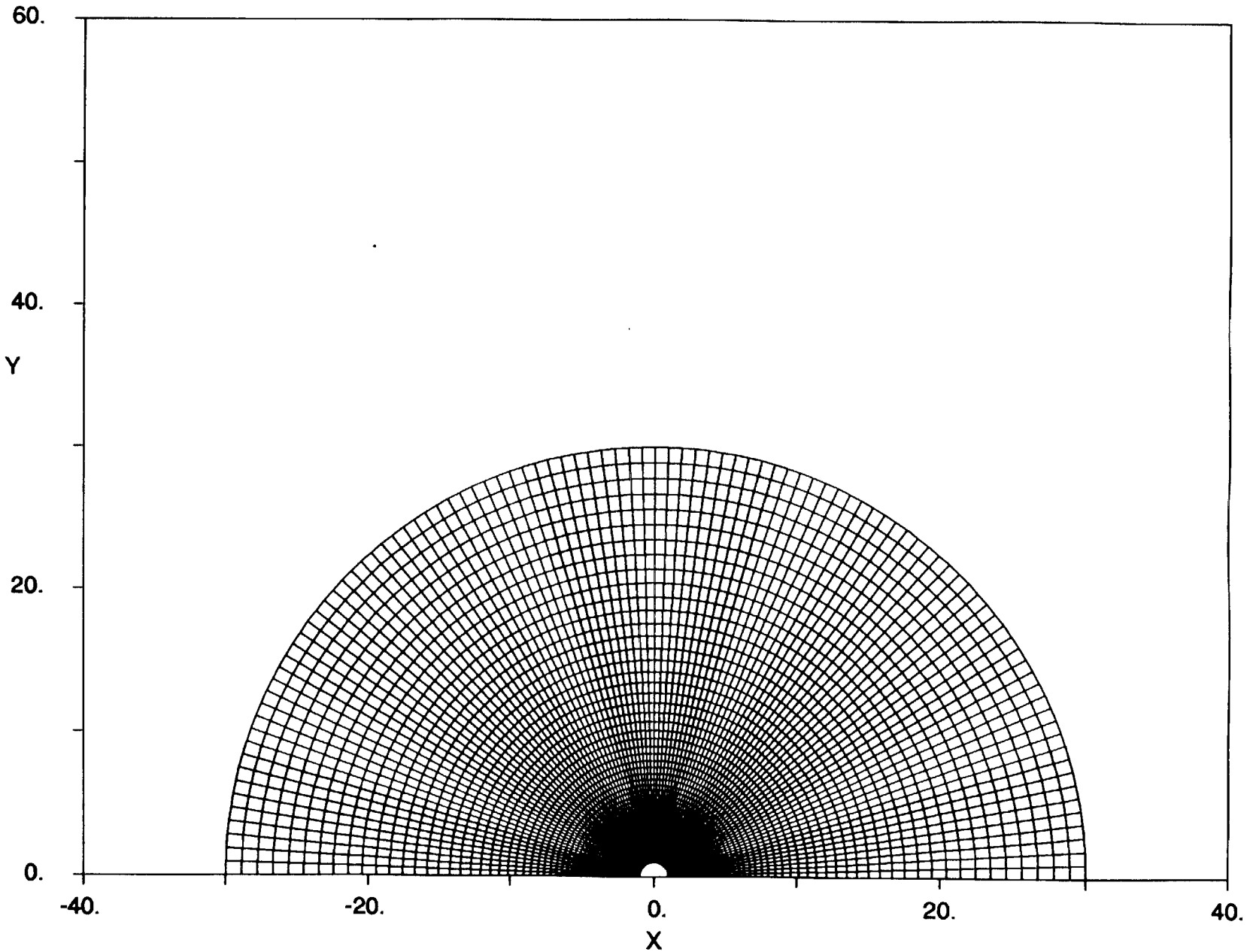


Fig. III.9 : Twice as Fine Grid for Viscous Flow Past a Circular Cylinder

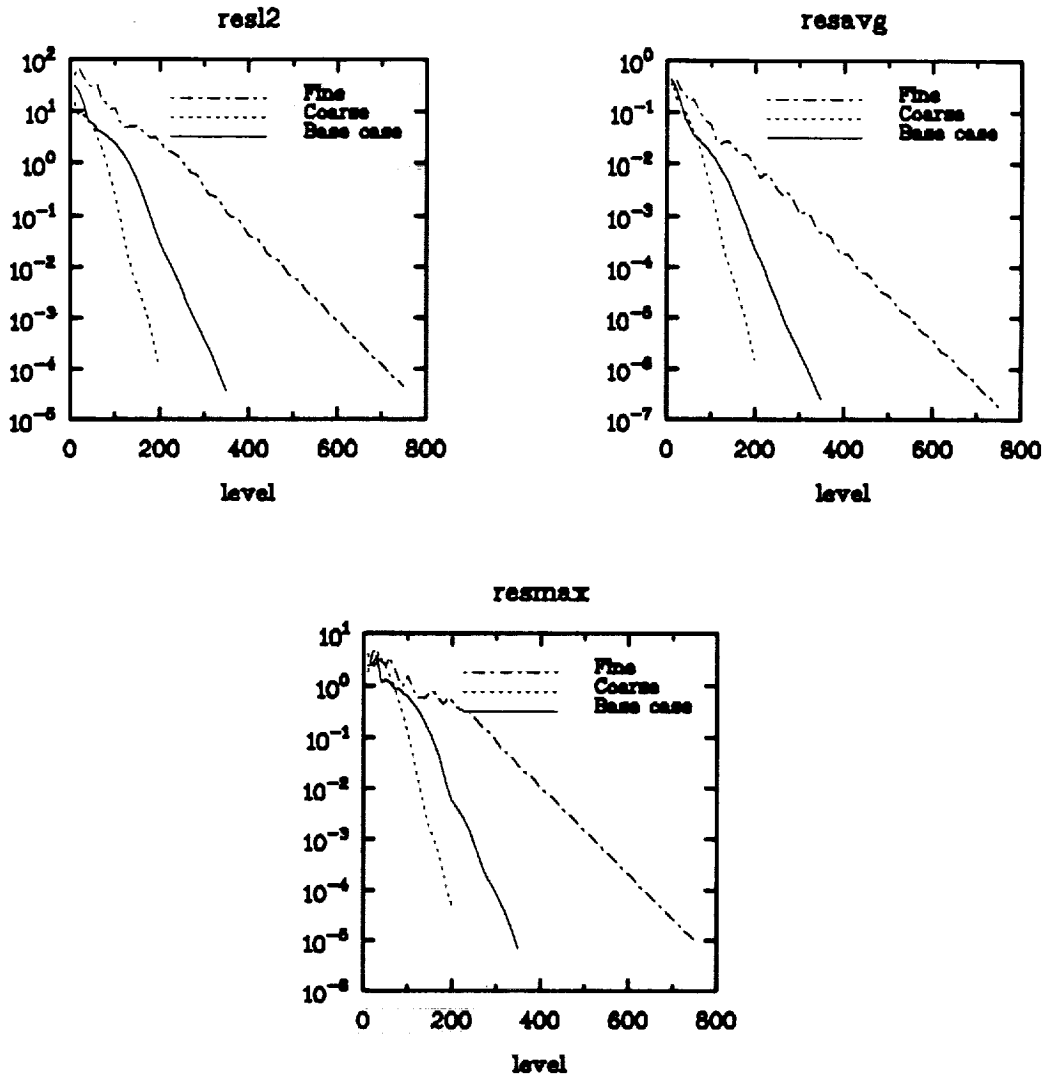


Fig. III.10 : Viscous flow past a circular cylinder
Convergence history for continuity equation
(Artificial Viscosity)
Base case grid 51•51

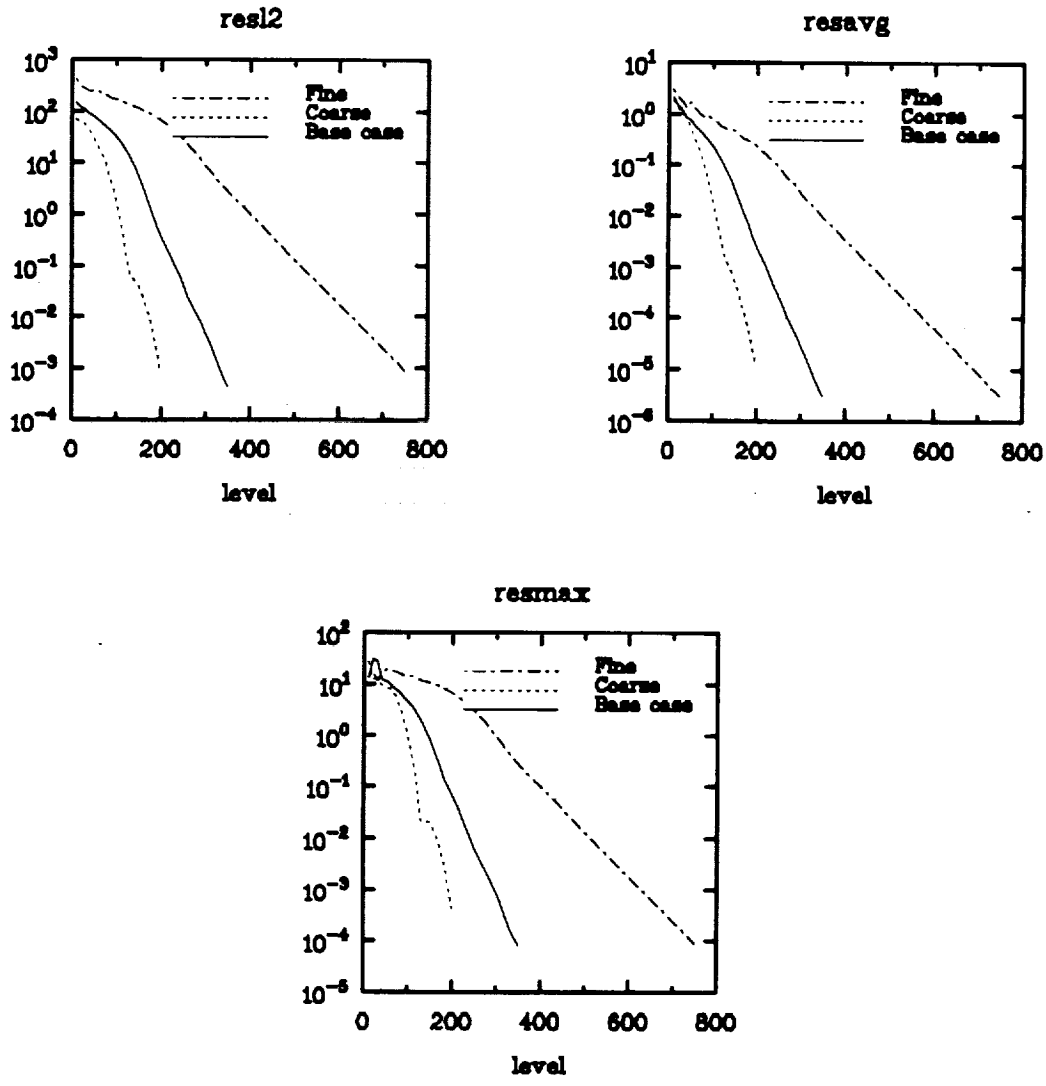


Fig. III.11 : Viscous flow past a circular cylinder
Convergence history for x-momentum equation
(Artificial Viscosity)
Base case grid 51*51

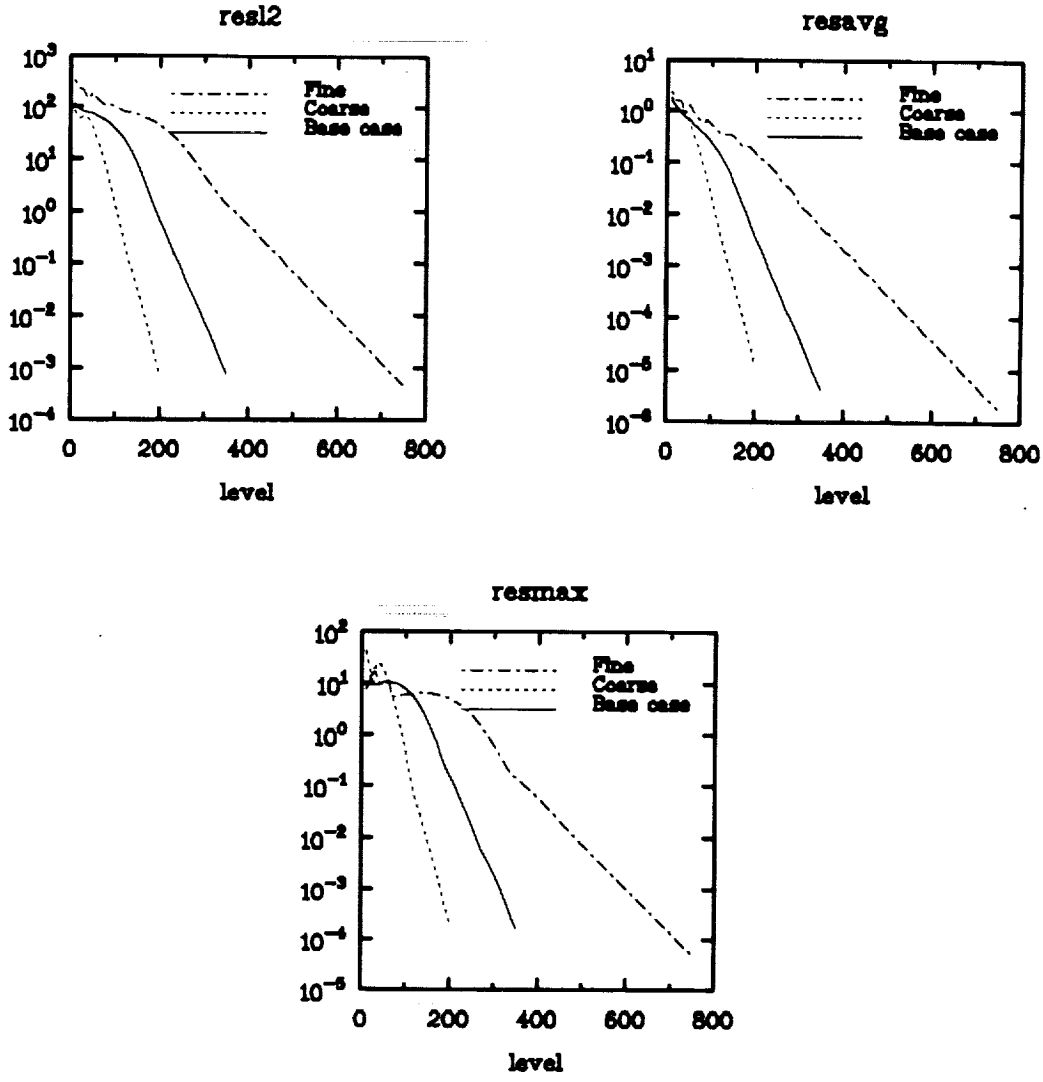


Fig. III.12 : Viscous flow past a circular cylinder
Convergence history for y-momentum equation
(Artificial Viscosity)
Base case grid 51•51

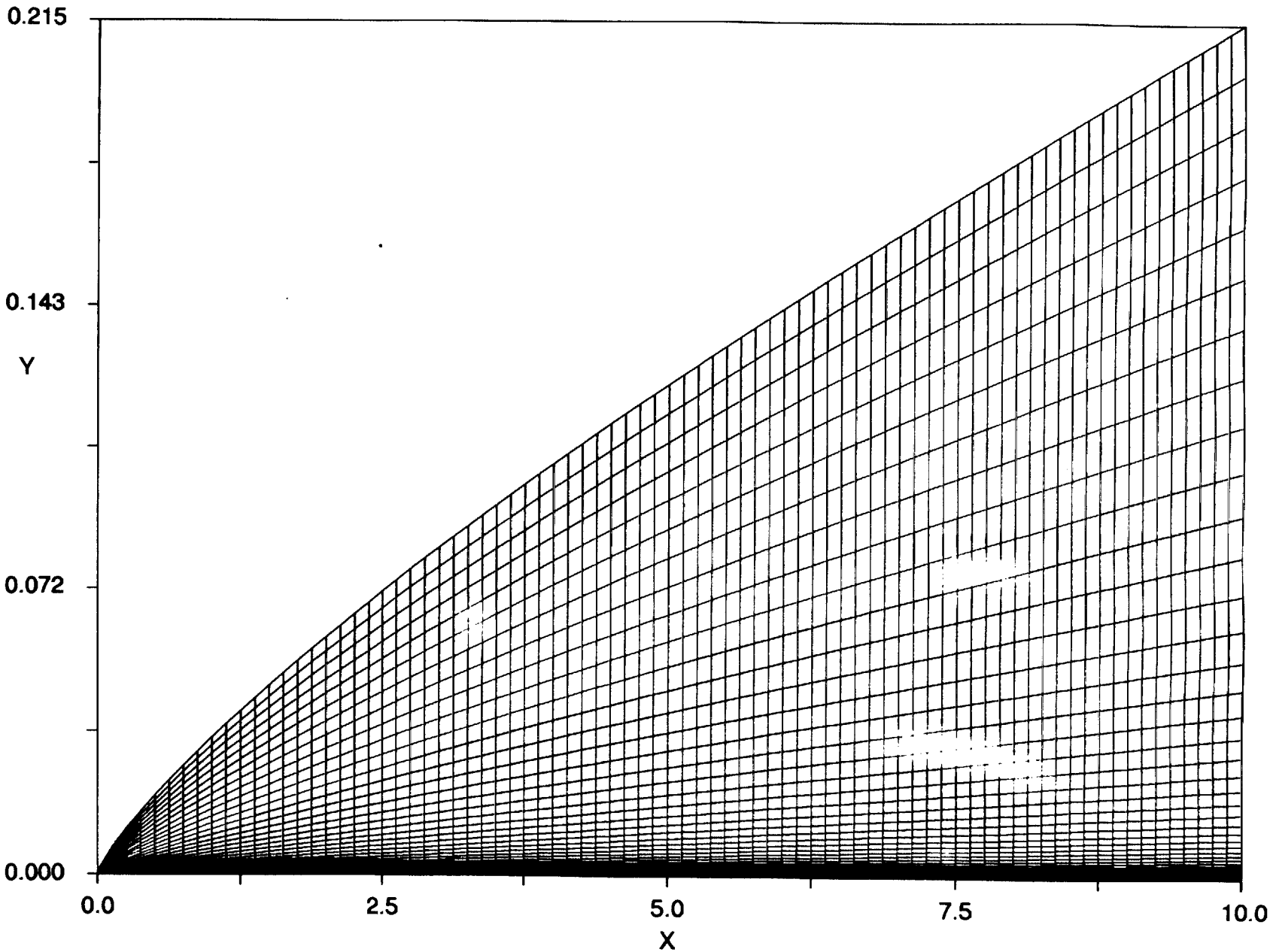


Fig. III.13 : Base-Case Grid for Flat Plate Turbulent Flow

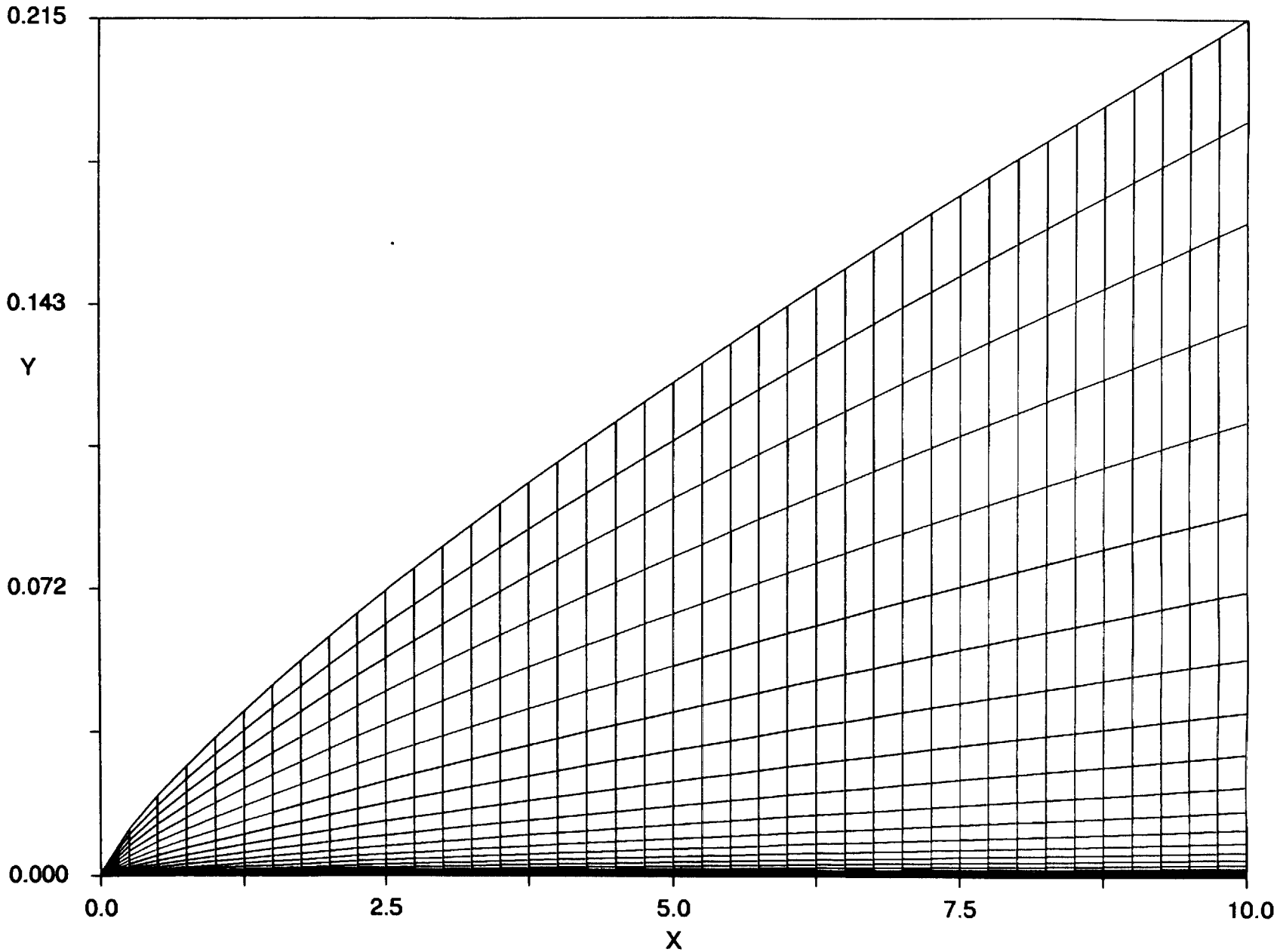


Fig. III.14 : Twice as Coarse Grid for Flat Plate Turbulent Flow

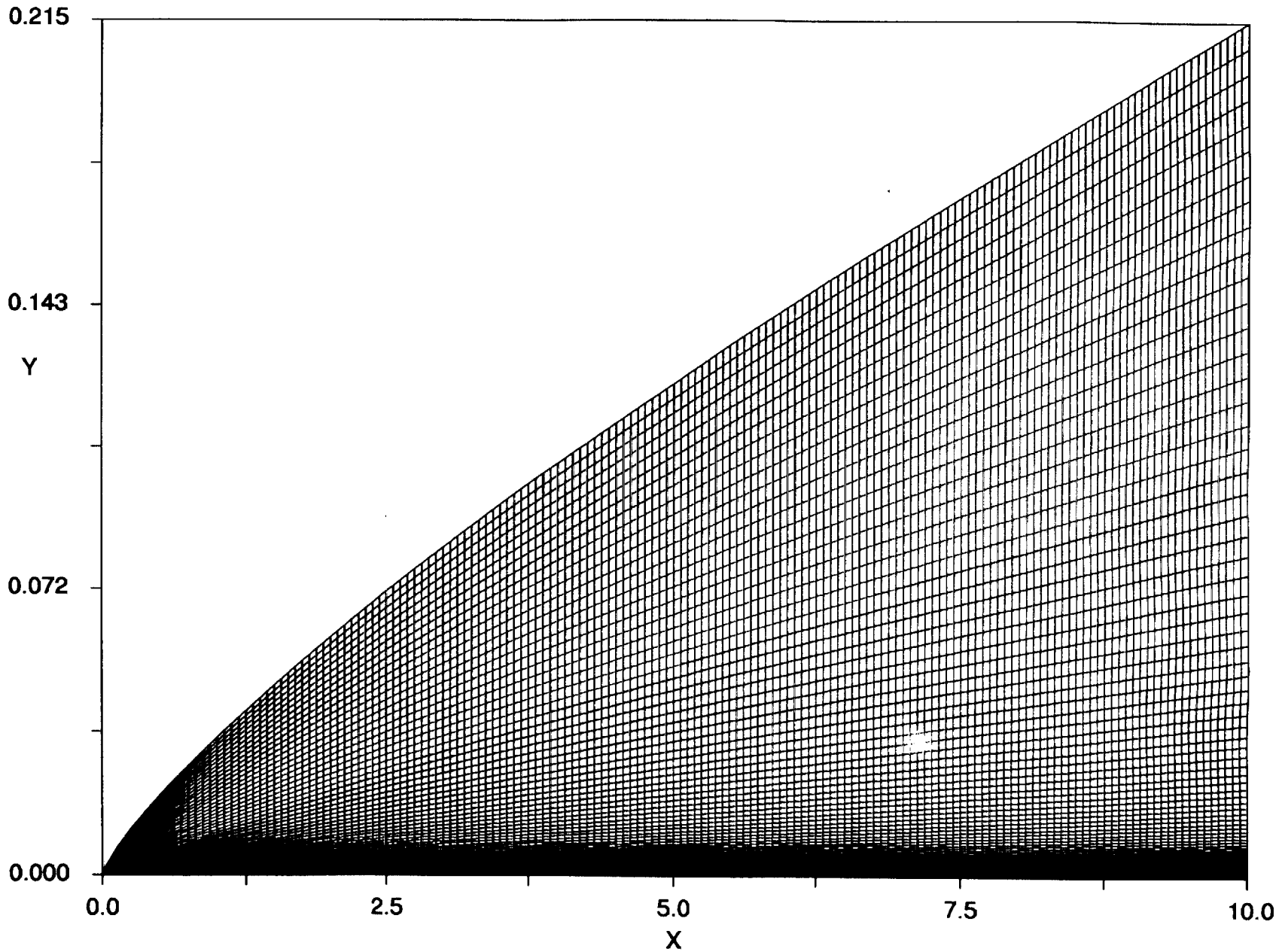


Fig. III.15 : Twice as Fine Grid for Flat Plate Turbulent Flow

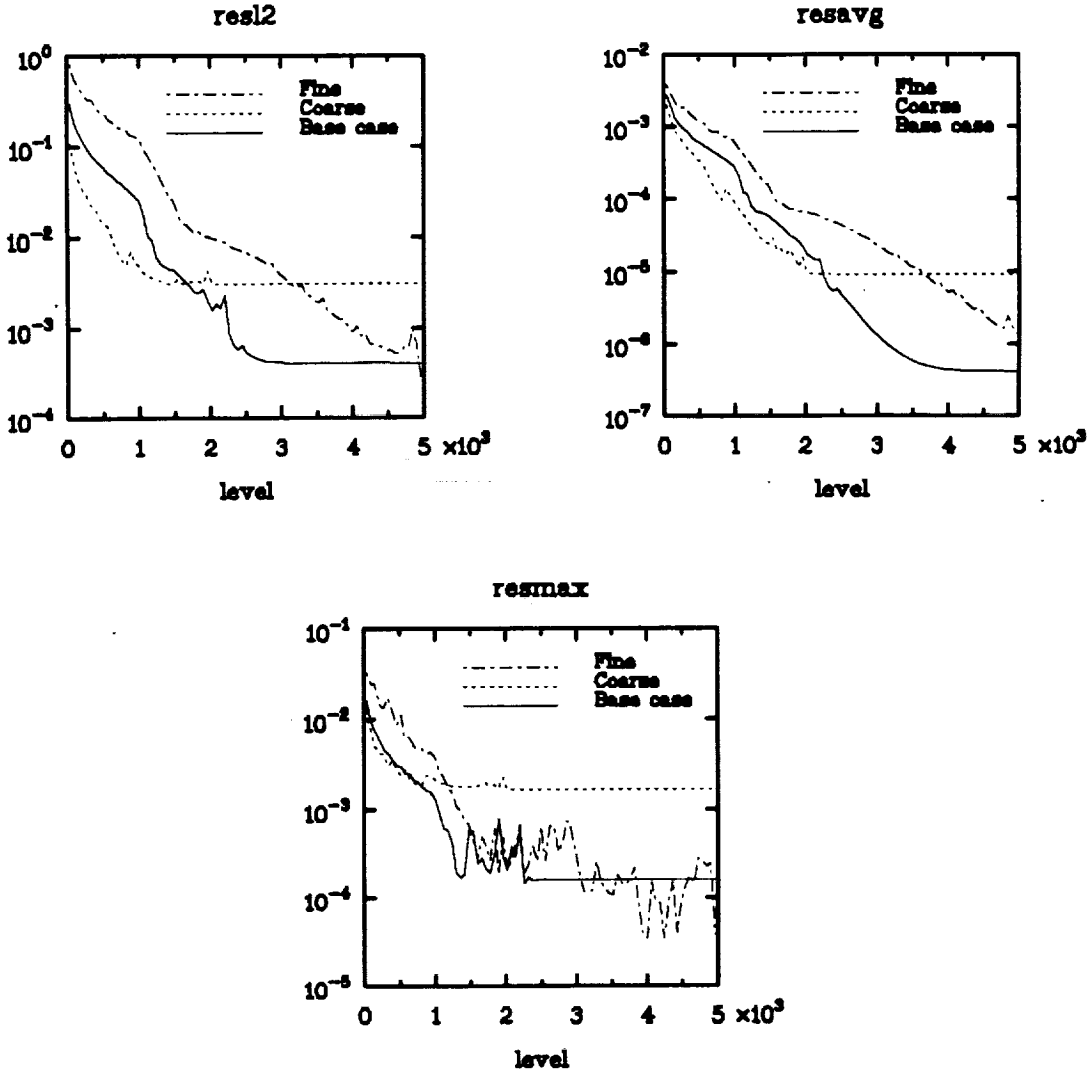


Fig. III.16 : Flat plate turbulent flow, Run 1
Convergence history for continuity equation
(Artificial Viscosity)
Base case grid 81*51

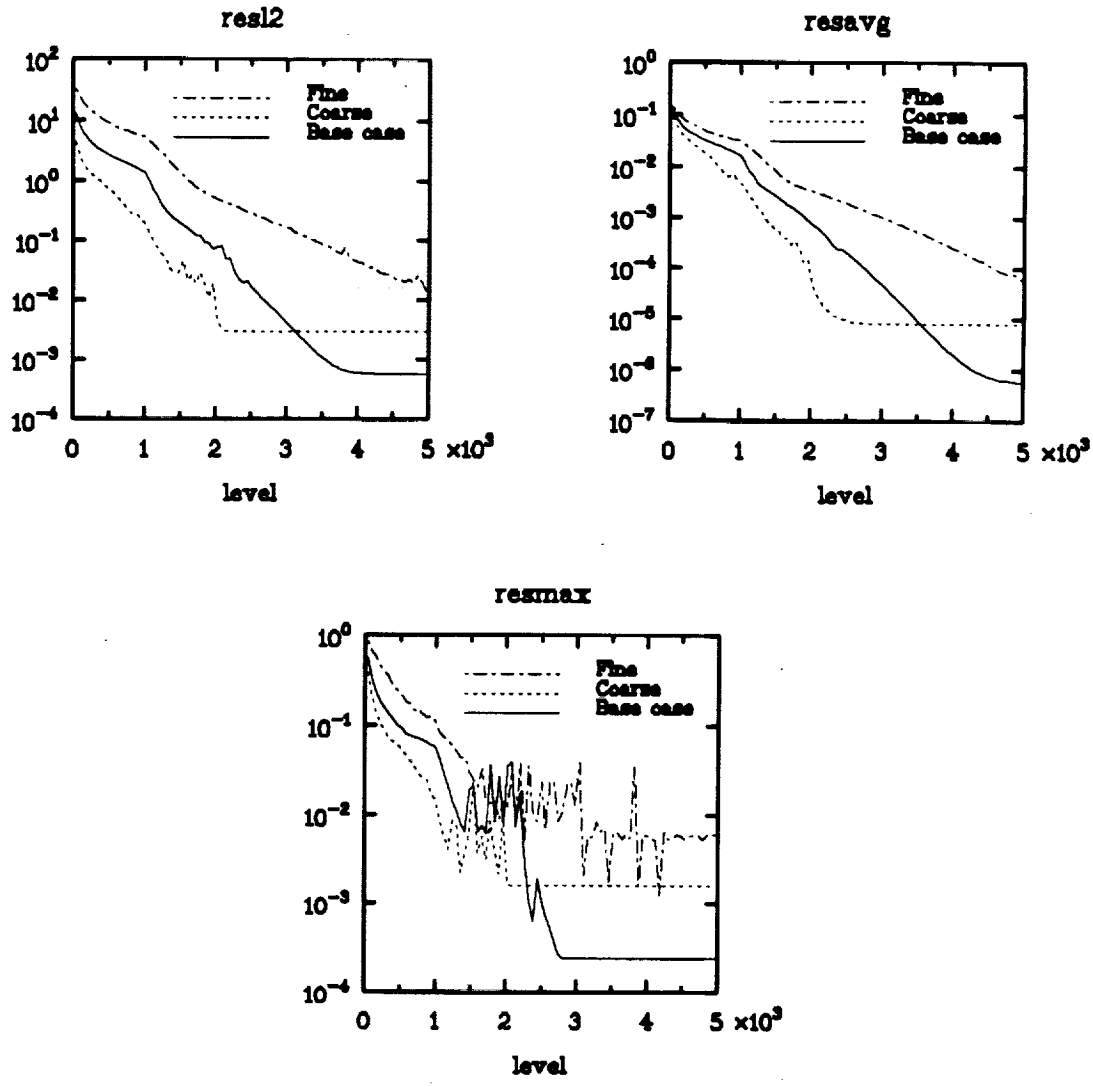


Fig. III.17 : Flat plate turbulent flow, Run 1
Convergence history for x-momentum equation
(Artificial Viscosity)
Base case grid 81*51

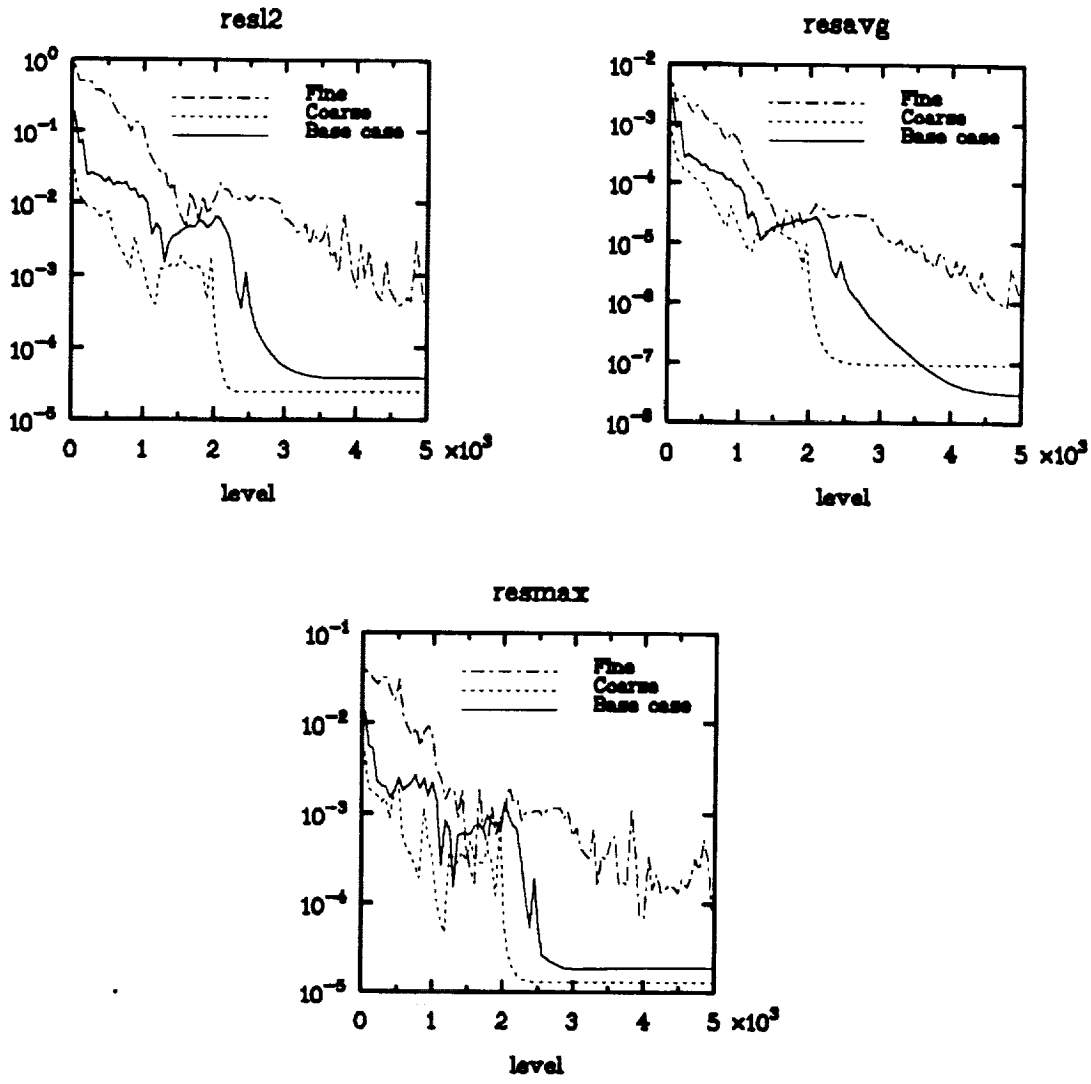


Fig. III.18 : Flat plate turbulent flow, Run 1
Convergence history for y-momentum equation
(Artificial Viscosity)
Base case grid 81*51

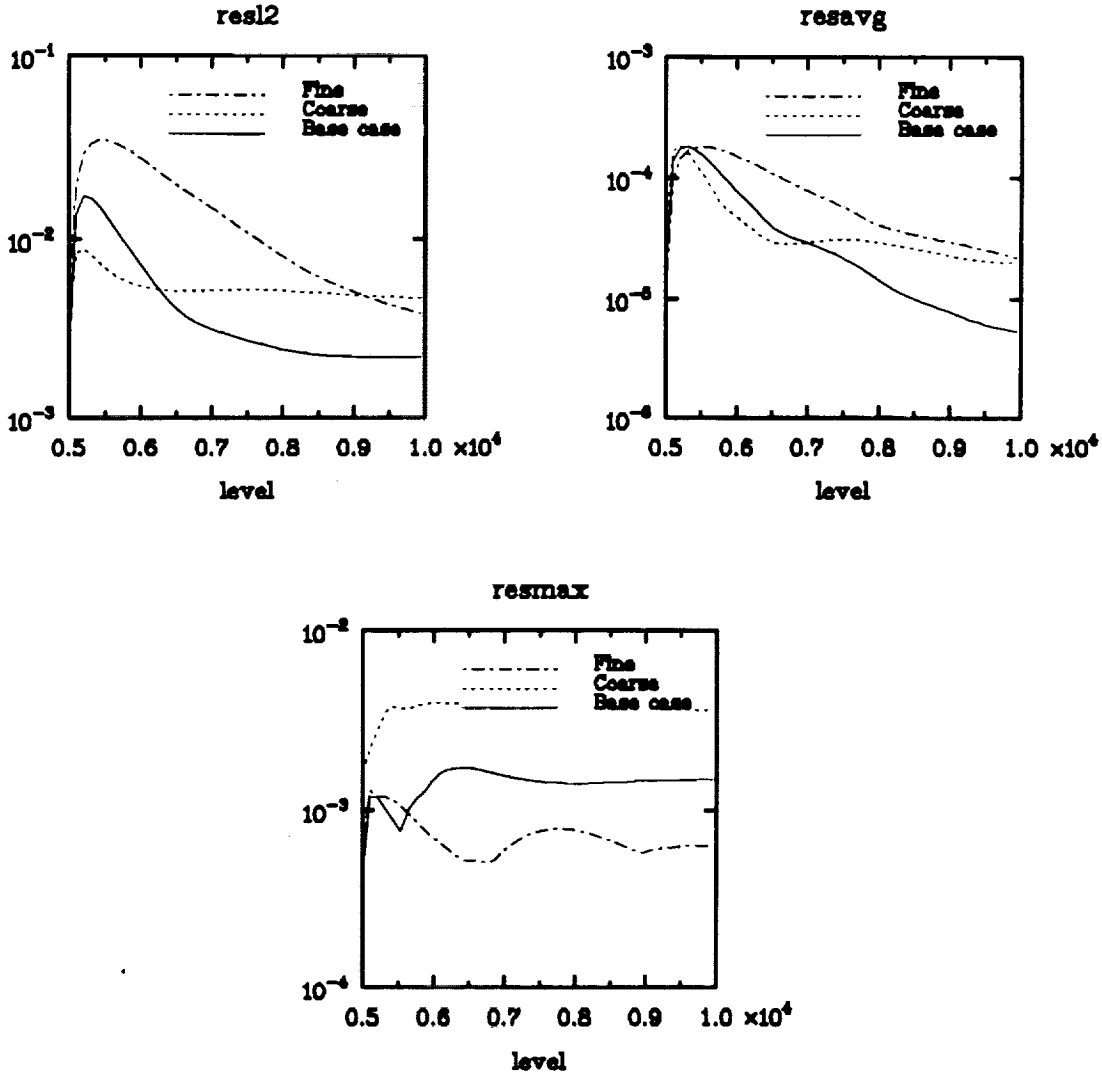


Fig. III.19 : Flat plate turbulent flow, Run 2
Convergence history for continuity equation
(Artificial Viscosity)
Base case grid 81*51

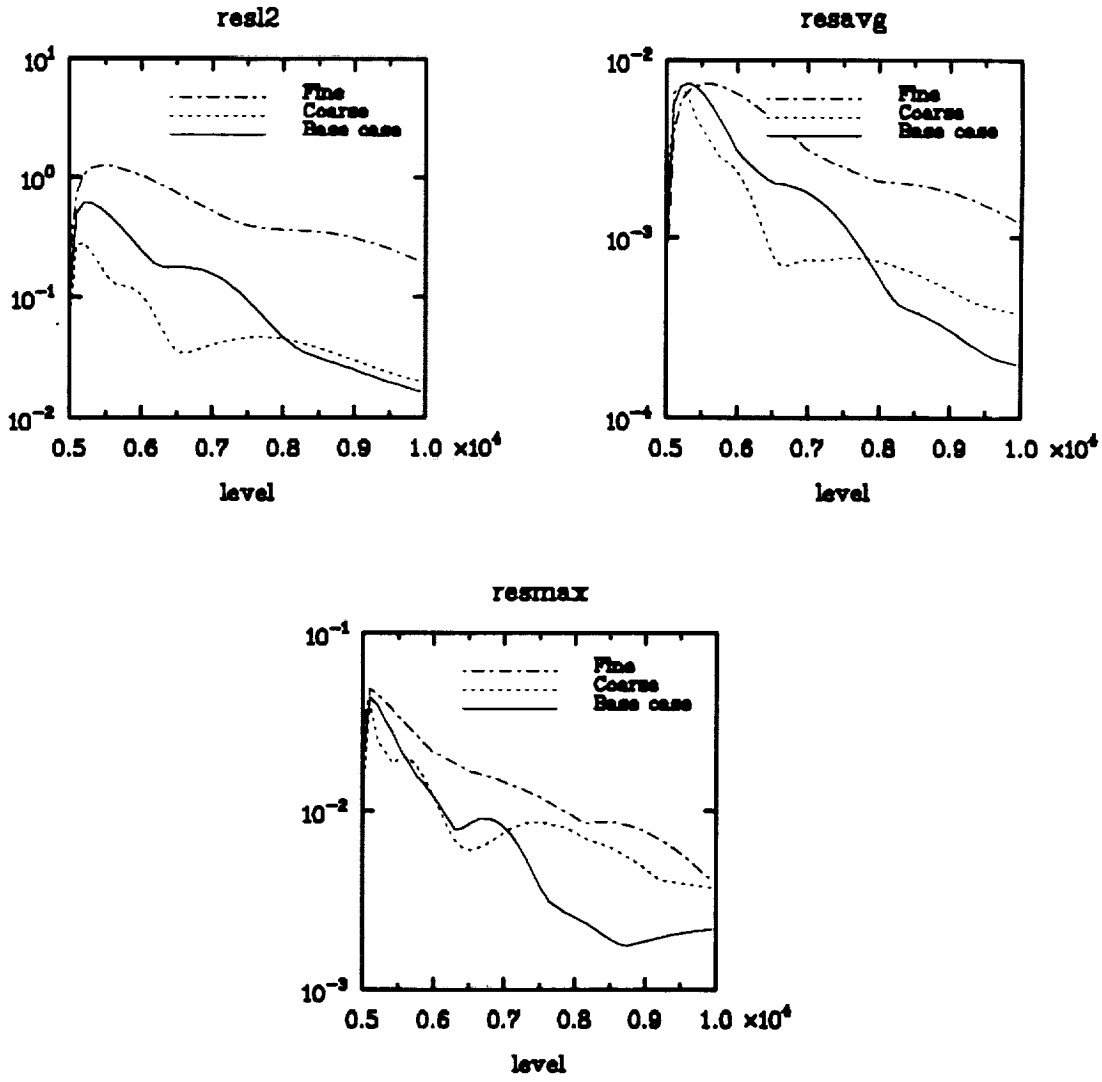


Fig. III.20 : Flat plate turbulent flow, Run 2
Convergence history for x-momentum equation
(Artificial Viscosity)
Base case grid 81*51

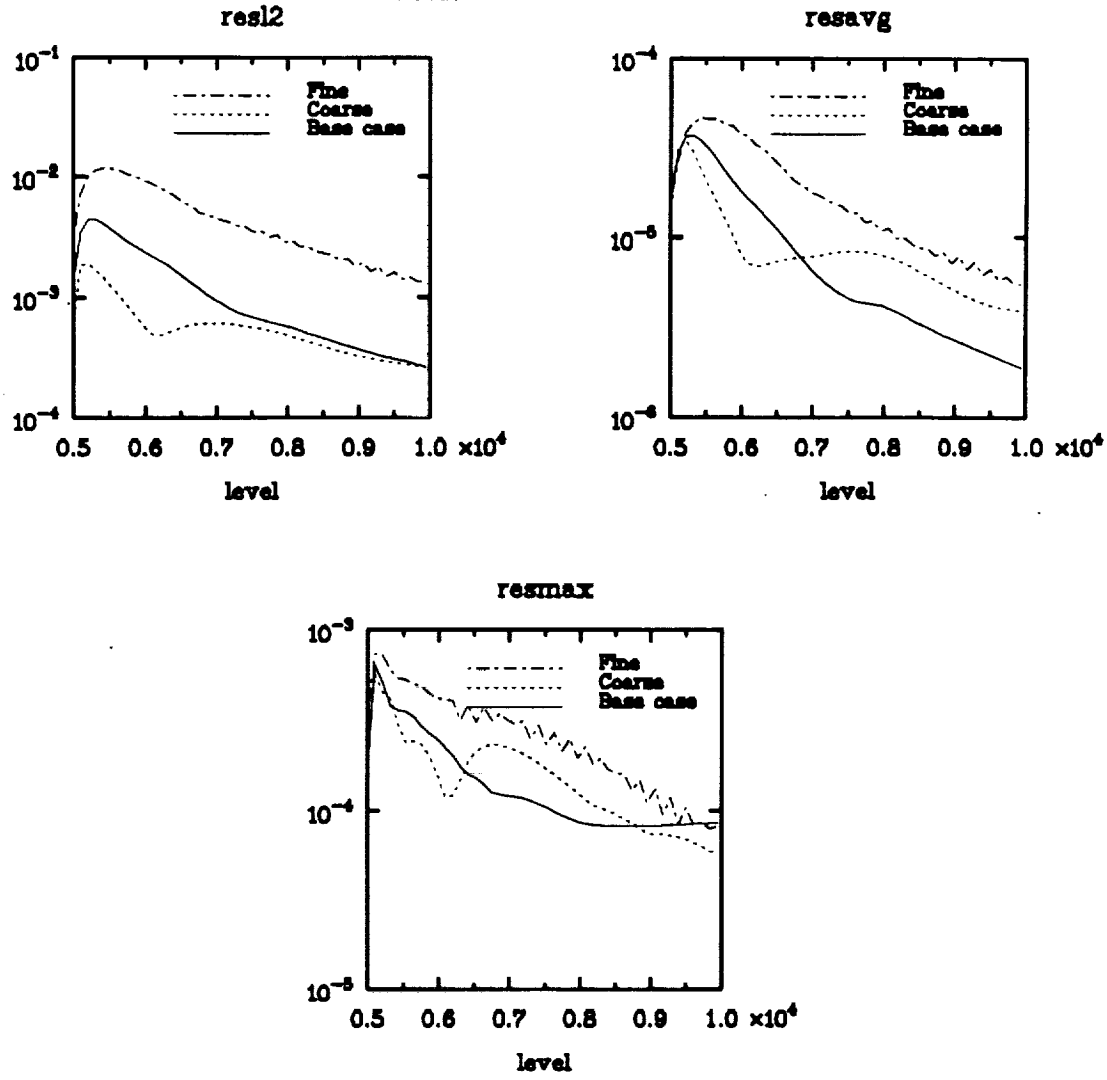


Fig. III.21 : Flat plate turbulent flow, Run 2
Convergence history for y-momentum equation
(Artificial Viscosity)
Base case grid 81*51

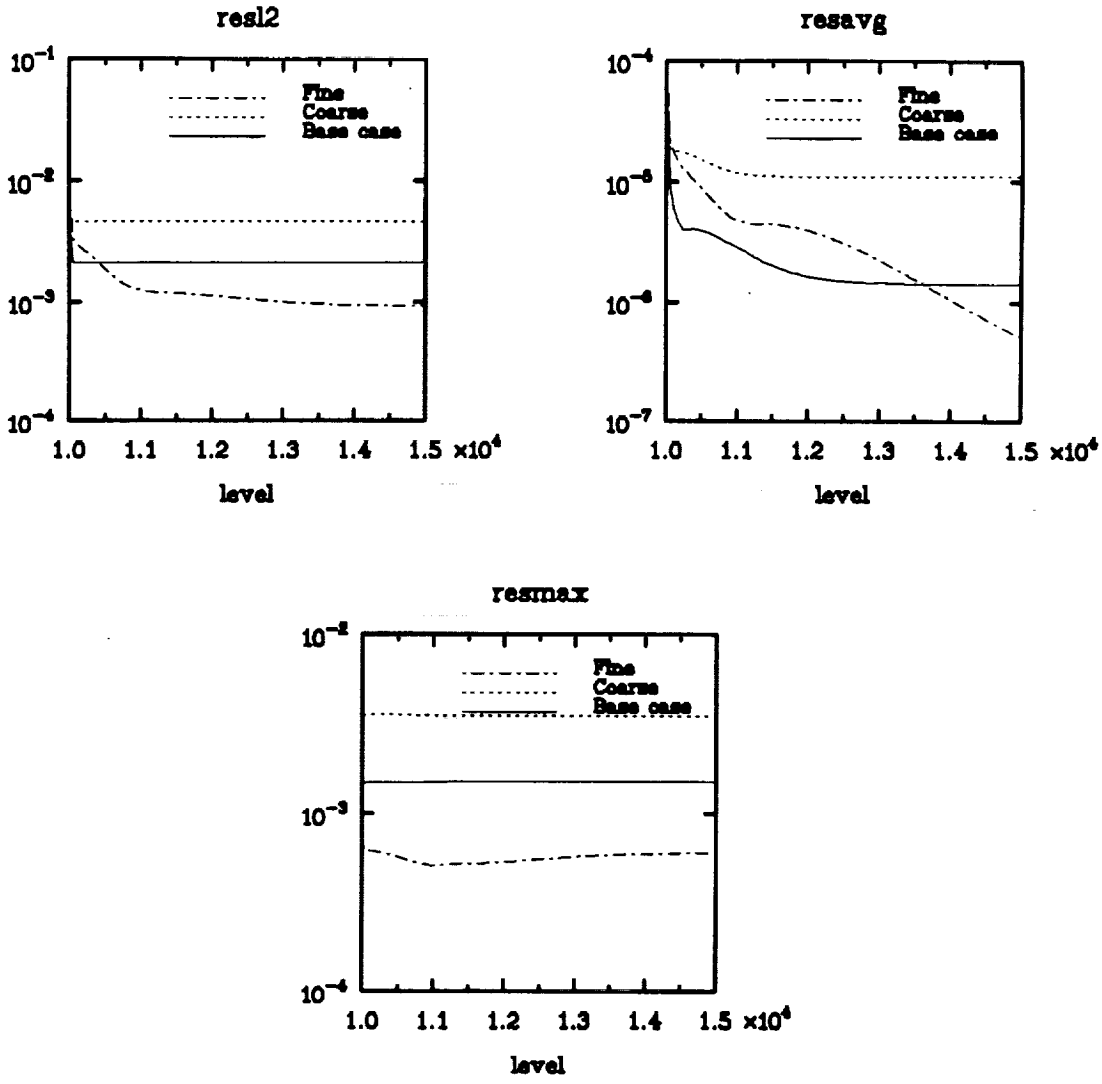


Fig. III.22 : Flat plate turbulent flow, Run 3
Convergence history for continuity equation
(Artificial Viscosity)
Base case grid 81*51

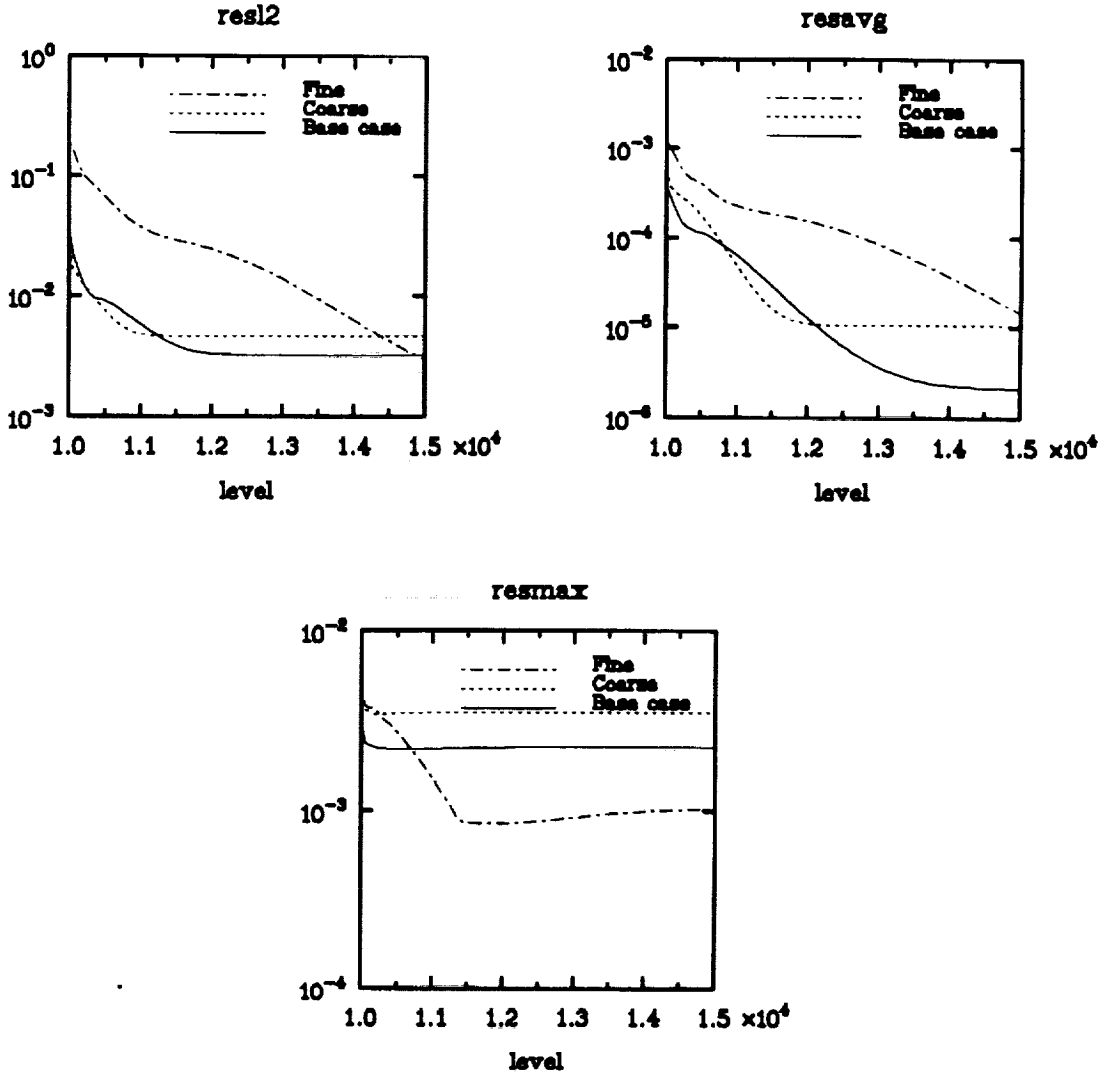


Fig. III.23 : Flat plate turbulent flow, Run 3
Convergence history for x-momentum equation
(Artificial Viscosity)
Base case grid 81*51

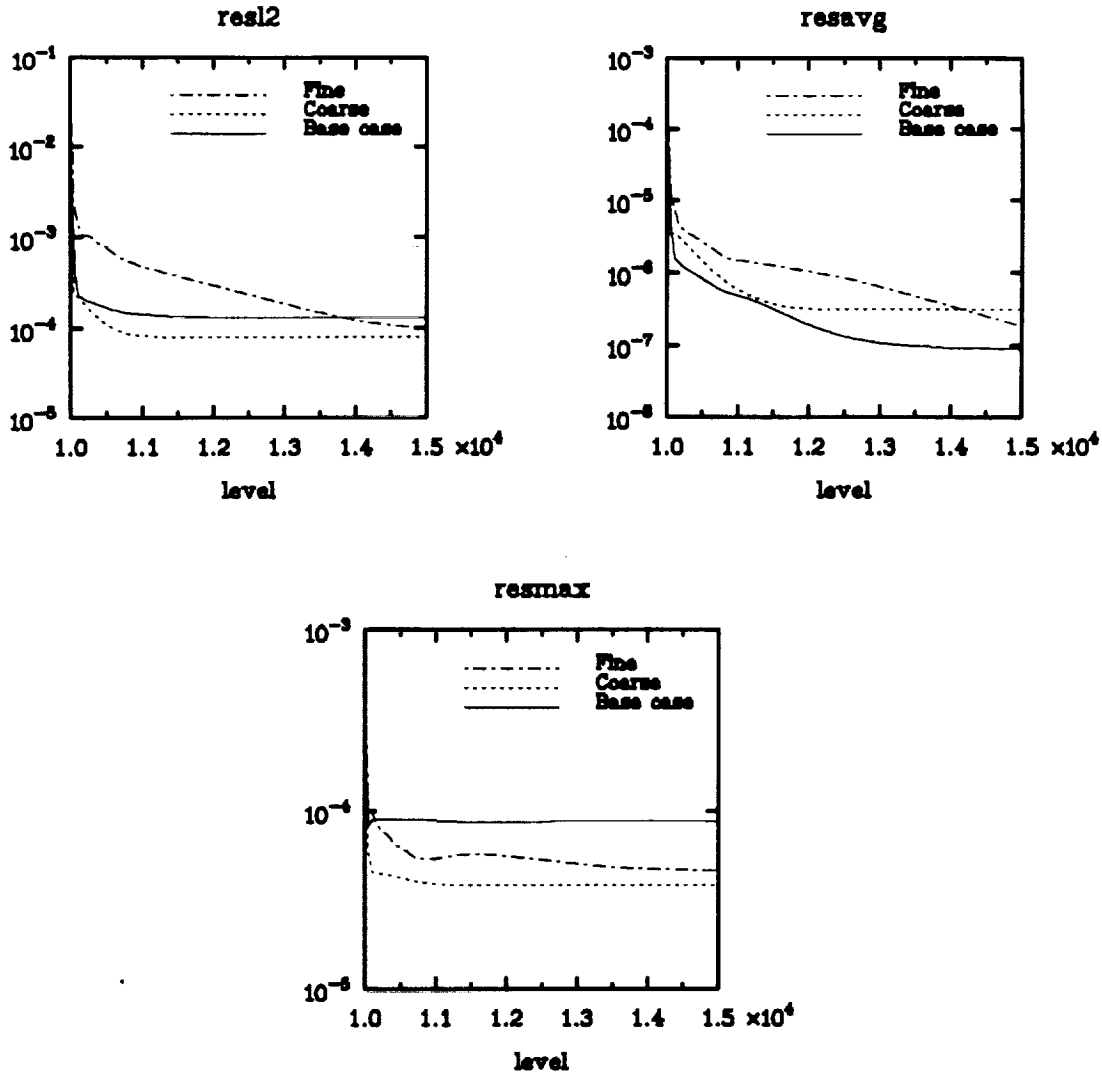


Fig. III.24 : Flat plate turbulent flow, Run 3
Convergence history for y-momentum equation
(Artificial Viscosity)
Base case grid 81*51

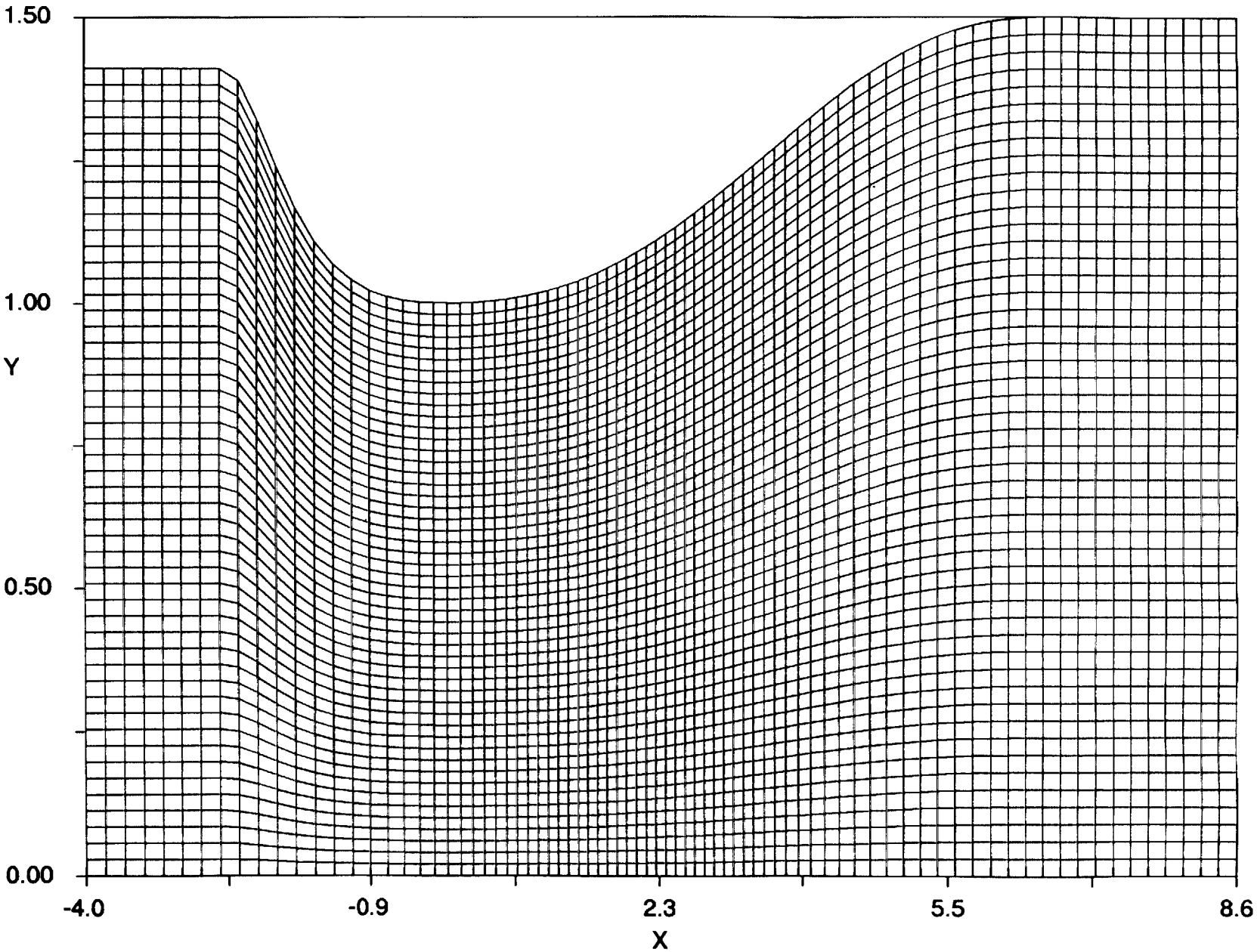


Fig. III.25 : Base-Case Grid for Sajben Transonic Flow

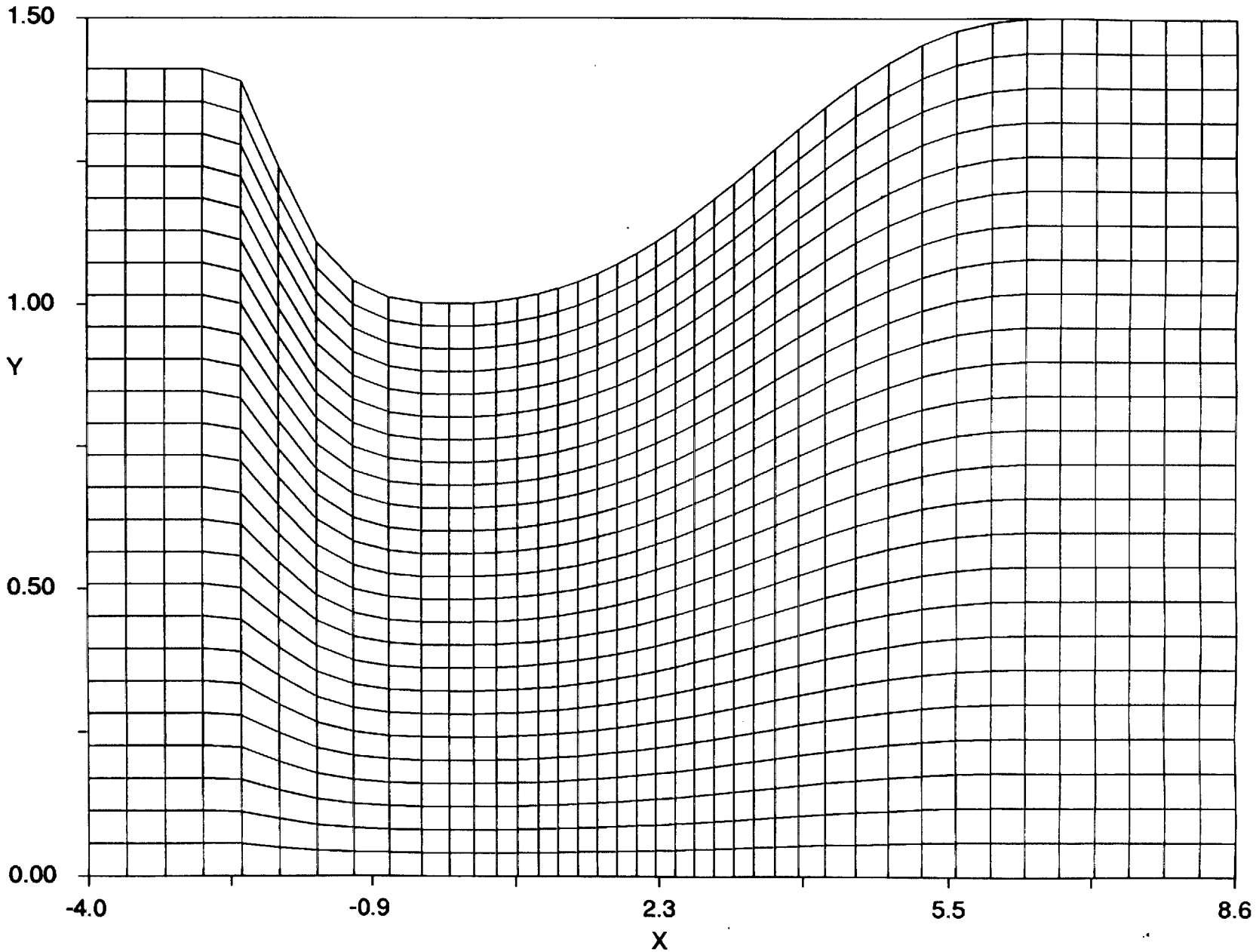


Fig. III.26 : Twice as Coarse Grid for Sajben Transonic Flow

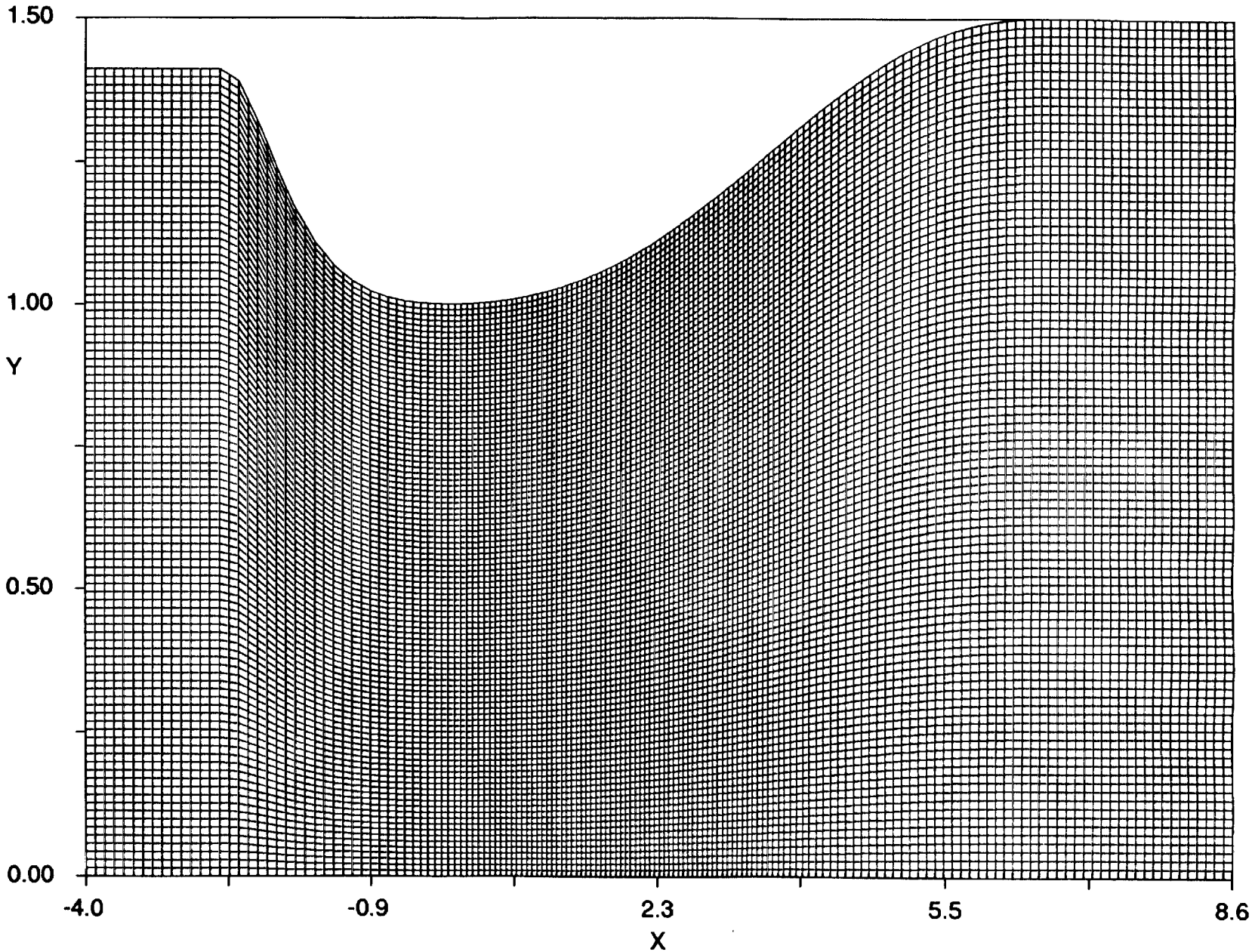


Fig. III.27 : Twice as Fine Grid for Sajben Transonic Flow

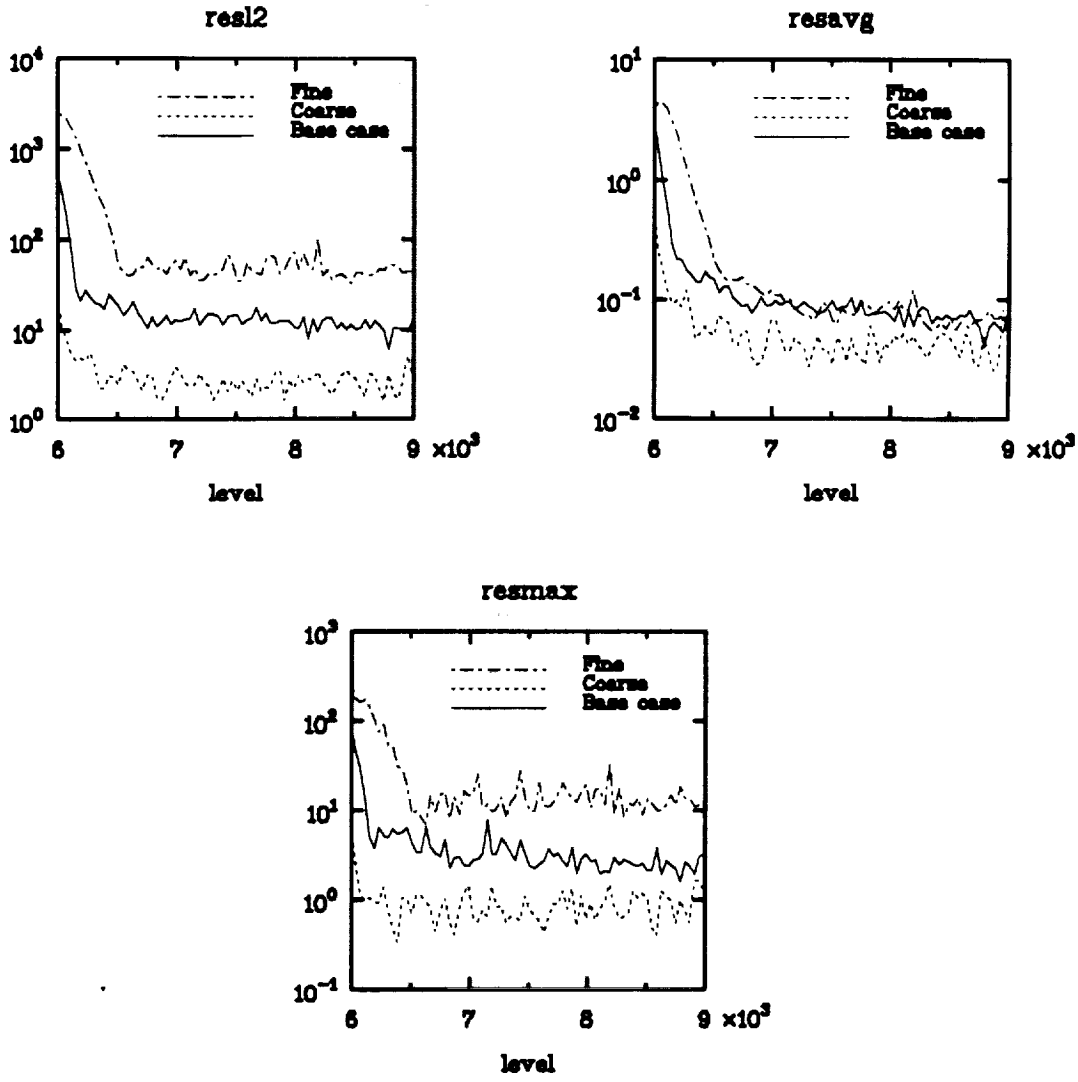


Fig. III.28 : Sajben transonic flow, Run 3
Convergence history for continuity equation
(Artificial Viscosity)
Base case grid 81*51

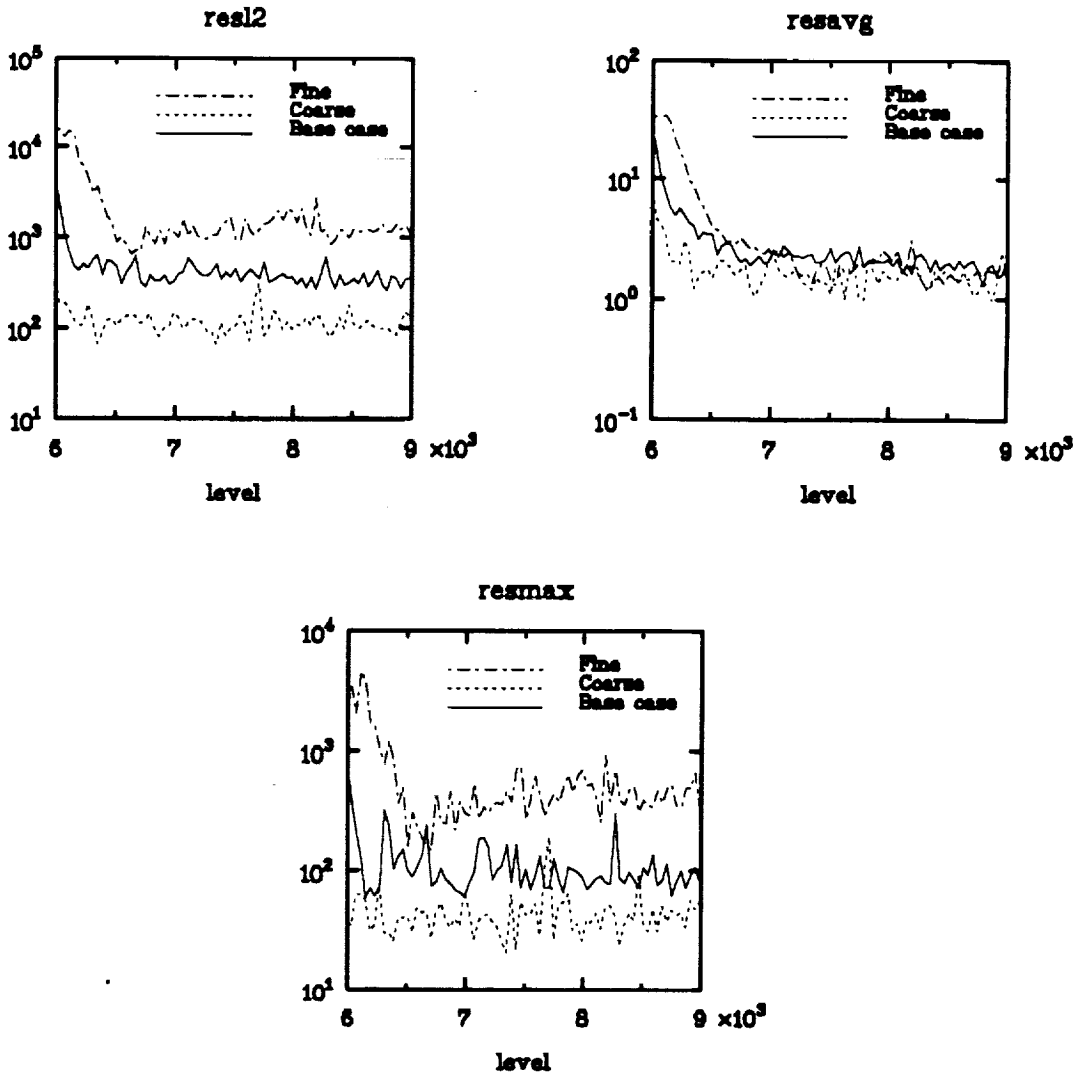


Fig. III.29 : Sajben transonic flow, Run 3
Convergence history for x-momentum equation
(Artificial Viscosity)
Base case grid 81*51

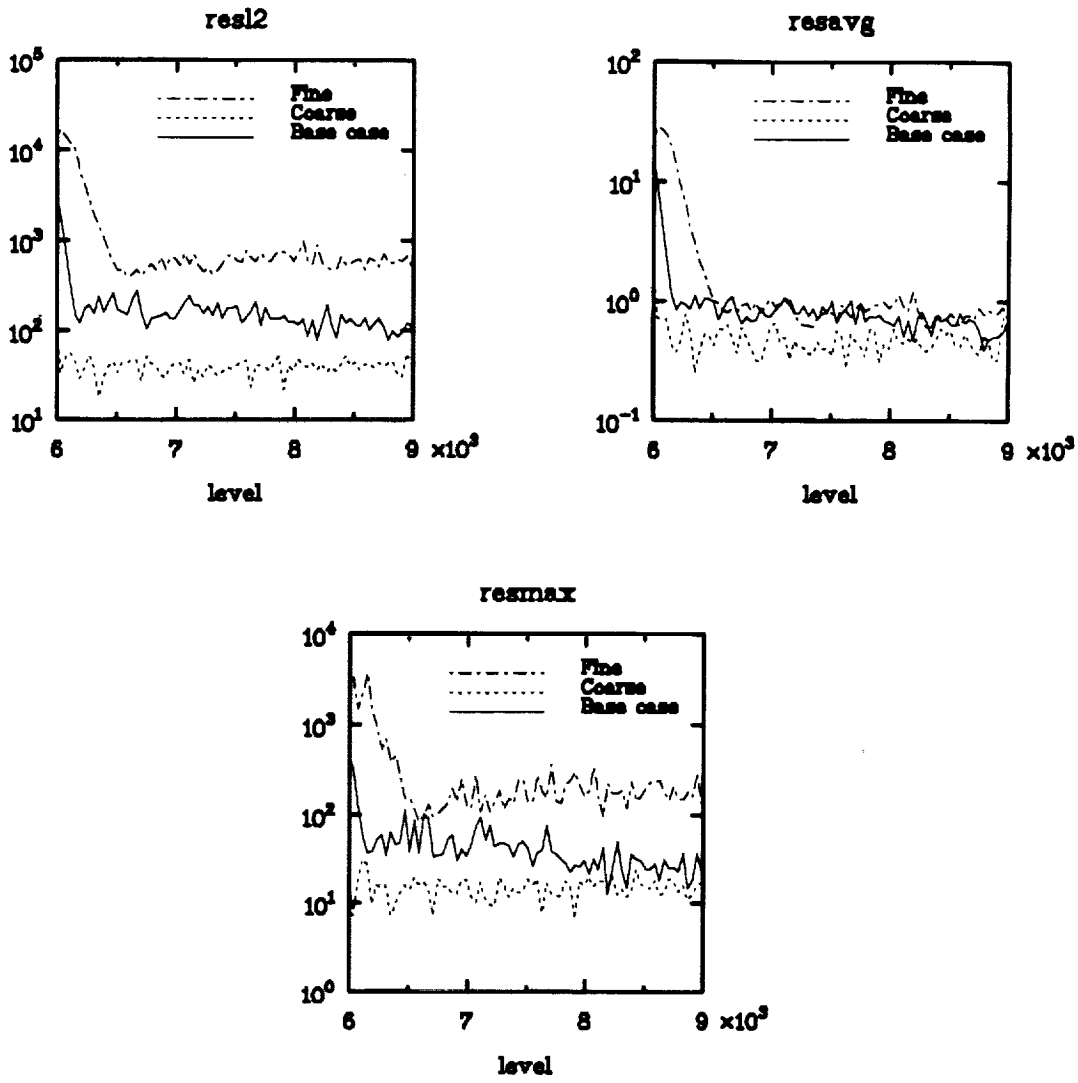


Fig. III.30 : Sajben transonic flow, Run 3
Convergence history for y-momentum equation
(Artificial Viscosity)
Base case grid 81*51

APPENDICES

Appendix A

Pseudo-code for the Steger and Warming Upwind Scheme for Euler Equation

Set up A^+, A^-, B^+, B^-, C^+, C^-

For CFL = 0, 50 do

For $\phi_x = 0, 2\pi$ *do*

For $\phi_y = 0, 2\pi$ *do*

For $\phi_z = 0, 2\pi$ *do*

Depending on split type, set up k and L

Solve generalized eigenvalue problem; 5 λ*'s*

Extract maximum eigenvalue

Compute Average

Extract λ_μ

end

end

end

Compute l_2 *norm*

Compute average of λ

Compute average of λ_{max}

end

Appendix B

Expressions for flux-vectors E, F and G.

$$\begin{aligned} Q &= [\rho, \rho u, \rho v, \rho w, \rho E_T]^T \\ &= [q_1,\ q_2,\ q_3,\ q_4,\ q_5]^T \end{aligned} \tag{B.1}$$

$$\begin{aligned} E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (\rho E_T + p)u \end{bmatrix} &= \begin{bmatrix} \rho u \\ \frac{(\rho u)^2}{\rho} + p \\ \frac{(\rho u)(\rho v)}{\rho} \\ \frac{(\rho u)(\rho w)}{\rho} \\ (\rho E_T + p)\frac{\rho u}{\rho} \end{bmatrix} = \begin{bmatrix} q_2 \\ \frac{q_2^2}{q_1} + p \\ \frac{q_2 q_3}{q_1} \\ \frac{q_2 q_4}{q_1} \\ (q_5 + p)\frac{q_2}{q_1} \end{bmatrix} \\ &= \begin{bmatrix} q_2 \\ \frac{q_2^2}{q_1} + (\gamma - 1)\left[q_5 - \frac{1}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right] \\ \frac{q_2 q_3}{q_1} \\ \frac{q_2 q_4}{q_1} \\ \left[\gamma q_5 - \frac{(\gamma-1)}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right]\frac{q_2}{q_1} \end{bmatrix} \end{aligned} \tag{B.2}$$

where

$$\begin{aligned} p &= (\gamma - 1)\left[\rho E_T - \frac{1}{2}\rho(u^2 + v^2 + w^2)\right] \\ &= (\gamma - 1)\left[\rho E_T - \frac{1}{2}\left(\frac{(\rho u)^2}{\rho} + \frac{(\rho v)^2}{\rho} + \frac{(\rho w)^2}{\rho}\right)\right] \\ &= (\gamma - 1)\left[q_5 - \frac{1}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right] \end{aligned} \tag{B.3}$$

Similar expressions for F and G in terms of the conserved variables are

$$F = \begin{bmatrix} \rho v \\ \rho uv \\ (\rho v^2 + p) \\ \rho vw \\ (\rho E_T + p)v \end{bmatrix} = \begin{bmatrix} q_3 \\ \frac{q_2 q_3}{q_1} \\ \frac{q_3^2}{q_1} + (\gamma - 1)\left[q_5 - \frac{1}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right] \\ \frac{q_3 q_4}{q_1} \\ \left[\gamma q_5 - \frac{(\gamma-1)}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right]\frac{q_3}{q_1} \end{bmatrix} \tag{B.4}$$

$$G = \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ (\rho w^2 + p) \\ (\rho E_T + p)w \end{bmatrix} = \begin{bmatrix} q_4 \\ \frac{q_2 q_4}{q_1} \\ \frac{q_3 q_4}{q_1} \\ \frac{q_4^2}{q_1} + (\gamma - 1)\left[q_5 - \frac{1}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right] \\ \left[\gamma q_5 - \frac{(\gamma-1)}{2}\left(\frac{q_2^2}{q_1} + \frac{q_3^2}{q_1} + \frac{q_4^2}{q_1}\right)\right]\frac{q_4}{q_1} \end{bmatrix} \tag{B.5}$$

Mathematica is employed to generate $A = \frac{\partial E}{\partial Q}$, $B = \frac{\partial F}{\partial Q}$, $C = \frac{\partial G}{\partial Q}$ and their respective splits using the above expressions. For example, the sequence of commands used in Mathematica to generate A and its splits is shown in *Appendix C*

Appendix C

Mathematica Sequence used to generate the Fortran code for splitting the flux-Jacobians A, B, C

```
q={q1,q2,q3,q4,q5}

e={ q2,

      q2^2/q1+(r-1)(q5-(1/2)(q2^2/q1+q3^2/q1+q4^2/q1)),

      q2 q3/q1,

      q2 q4/q1,

      (r q5 -((r-1)/2)(q2^2/q1+q3^2/q1+q4^2/q1))(q2/q1)

    }

a=Factor[ Table[ D[ e[[i]], q[[j]] ],{i,5},{j,5}] ]

q1=rho

q2=rho u

q3=rho v

q4=rho w

q5=rho h

h=c^2/(r(r-1))+(u^2+v^2+w^2)/2

a=Factor[a]

xs={d,x}=Factor[Eigensystem[a]] x1=x[[1]]/x[[1,1]]

x2=x[[2]]/x[[2,1]]
```

```
x3=x[[3]]/x[[3,1]]

x4=x[[4]]/x[[4,1]] x5=x[[5]]/x[[5,1]] x={x1,x2,x3,x4,x5}

xt=Factor[Transpose[x]]

dd=DiagonalMatrix[d]

xi=Factor[Inverse[xt]]

ident=Factor[xt.xi]

aa=Factor[xt.dd.xi]

rhs=Factor[xt.dd]

lhs=Factor[a.xt]

zero=Factor[rhs-lhs] ident1=Factor[Inverse[rhs].lhs]

dp=DiagonalMatrix[{u,u,u,0,c+u}] dm=DiagonalMatrix[{0,0,0,-c + u,0}] ap=Factor[xt.dp.xi]

am=Factor[xt.dm.xi]

asum=Factor[ap+am]

zero2=Factor[aa-asum]

Do[

    ToString[

        StringForm[

            "ap(‘‘,‘‘)=‘‘",i,j,FortranForm[ap[[i,j]]]

]

        ] >>>xxx,
```

```
        {i,5},{j,5}
    ]
Do[
    ToString[
                StringForm[
                        "am(‘‘,‘‘)=‘‘",i,j,FortranForm[am[[i,j]]]
                    ]
                ] >>>xxx,
        {i,5},{j,5}
    ]
```

Appendix D

Pseudo-code for the Beam and Warming Central Scheme for Euler Equations

Set up A, B, C

For CFL = 0, 50 do

For $\phi_x = 0, 2\pi$ *do*

For $\phi_y = 0, 2\pi$ *do*

For $\phi_z = 0, 2\pi$ *do*

Set up k and L

Solve generalized eigenvalue problem; 5 λ*'s*

Extract maximum eigenvalue

Compute Average

Extract λ_μ

end

end

end

Compute l_2 *norm*

Compute average of λ

Compute average of λ_{max}

end

Appendix E

Pseudo-code for the Beam and Warming Central Scheme for Navier-Stokes Equations

Set up A, B, C, R, R_1, R_2, S, S_1, S_2, Y, Y_1, Y_2

For CFL = 0, 50 do

For ϕ_x = 0, 2π do

For ϕ_y = 0, 2π do

For ϕ_z = 0, 2π do

Set up k and L

Solve generalized eigenvalue problem; 5 λ's

Extract maximum eigenvalue

Compute Average

Extract λ_μ

end

end

end

Compute l_2 norm

Compute average of λ

Compute average of λ_{max}

end

Appendix F

Expressions for U_1, U_2, U_3, V_1, V_2, V_3, W_1, W_2, W_3 in the Analysis of Navier-Stokes Equations

$$\begin{aligned} Q &= [\rho, \rho u, \rho v, \rho w, \rho E_T]^T \\ &= [q_1,\ q_2,\ q_3,\ q_4,\ q_5]^T \end{aligned} \tag{F.1}$$

$$\begin{aligned} Q_x &= [\rho_x,\ (\rho u)_x,\ (\rho v)_x,\ (\rho w)_x,\ (\rho E_T)_x]^T \\ &= [m_1\ , m_2, m_3\ , m_4\ , m_5]^T \end{aligned}$$

$$\begin{aligned} Q_y &= \left[\rho_y,\ (\rho u)_y,\ (\rho v)_y,\ (\rho w)_y,\ (\rho E_T)_y\right]^T \\ &= [n_1\ , n_2, n_3\ , n_4\ , n_5]^T \end{aligned} \tag{F.2}$$

$$\begin{aligned} Q_z &= [\rho_z,\ (\rho u)_z,\ (\rho v)_z,\ (\rho w)_z,\ (\rho E_T)_z]^T \\ &= [l_1,\ l_2,\ l_3,\ l_4,\ l_5]^T \end{aligned}$$

$$\begin{aligned} u_x &= \left(\frac{\rho u}{\rho}\right)_x = \left(\frac{q_2}{q_1}\right)_x = \frac{q_1 q_{2x} - q_2 q_{1x}}{q_1^2} = \frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2} \\ v_x &= \left(\frac{\rho v}{\rho}\right)_x = \left(\frac{q_3}{q_1}\right)_x = \frac{q_1 q_{3x} - q_3 q_{1x}}{q_1^2} = \frac{m_3}{q_1} - \frac{q_3 m_1}{q_1^2} \\ w_x &= \left(\frac{\rho w}{\rho}\right)_x = \left(\frac{q_4}{q_1}\right)_x = \frac{q_1 q_{4x} - q_4 q_{1x}}{q_1^2} = \frac{m_4}{q_1} - \frac{q_4 m_1}{q_1^2} \end{aligned} \tag{F.3}$$

Similarly

$$
\begin{aligned}
u_y &= \frac{n_2}{q_1} - \frac{q_2 n_1}{q_1^2} \\
v_y &= \frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2} \\
w_y &= \frac{n_4}{q_1} - \frac{q_4 n_1}{q_1^2}
\end{aligned}
\tag{F.4}
$$

and

$$
\begin{aligned}
u_z &= \frac{l_2}{q_1} - \frac{q_2 l_1}{q_1^2} \\
v_z &= \frac{l_3}{q_1} - \frac{q_3 l_1}{q_1^2} \\
w_z &= \frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}
\end{aligned}
\tag{F.5}
$$

Also

$$
\begin{aligned}
T_x &= \left\{ \frac{1}{\rho c_v} \left[\rho E_T - \frac{1}{2}(\rho u^2 + \rho v^2 + \rho w^2) \right] \right\}_x \quad since \quad \rho E_T = \rho c_v T + \frac{1}{2}\rho(u^2 + v^2 + w^2) \\
&= \frac{1}{c_v} \left[\frac{\rho E_T}{\rho} - \frac{1}{2}\frac{(\rho u)^2}{\rho^2} - \frac{1}{2}\frac{(\rho v)^2}{\rho^2} - \frac{1}{2}\frac{(\rho w)^2}{\rho^2} \right]_x \\
&= \frac{1}{c_v} \left[\left(\frac{q_5}{q_1} \right)_x - \frac{1}{2}\left(\frac{q_2^2}{q_1^2} \right)_x - \frac{1}{2}\left(\frac{q_3^2}{q_1^2} \right)_x - \frac{1}{2}\left(\frac{q_4^2}{q_1^2} \right)_x \right] \\
&= \frac{1}{c_v} [\frac{q_1 q_{5x} - q_5 q_{1x}}{q_1^2} - \frac{1}{2}\left(\frac{2q_1^2 q_2 q_{2x} - 2q_2^2 q_1 q_{1x}}{q_1^4} \right) - \\
&\qquad \frac{1}{2}\left(\frac{2q_1^2 q_3 q_{3x} - 2q_3^2 q_1 q_{1x}}{q_1^4} \right) - \frac{1}{2}\left(\frac{2q_1^2 q_4 q_{4x} - 2q_4^2 q_1 q_{1x}}{q_1^4} \right)] \\
&= \frac{1}{c_v} \left[\frac{m_5}{q_1} - \frac{q_5 m_1}{q_1^2} - \frac{q_2 m_2}{q_1^2} + \frac{q_2^2 m_1}{q_1^3} - \frac{q_3 m_3}{q_1^2} + \frac{q_3^2 m_1}{q_1^3} - \frac{q_4 m_4}{q_1^2} + \frac{q_4^2 m_1}{q_1^3} \right]
\end{aligned}
\tag{F.6}
$$

Similarly

$$T_y = \frac{1}{c_v}\left[\frac{n_5}{q_1} - \frac{q_5 n_1}{q_1^2} - \frac{q_2 n_2}{q_1^2} + \frac{q_2^2 n_1}{q_1^3} - \frac{q_3 n_3}{q_1^2} + \frac{q_3^2 n_1}{q_1^3} - \frac{q_4 n_4}{q_1^2} + \frac{q_4^2 n_1}{q_1^3}\right]$$
$$T_z = \frac{1}{c_v}\left[\frac{l_5}{q_1} - \frac{q_5 l_1}{q_1^2} - \frac{q_2 l_2}{q_1^2} + \frac{q_2^2 l_1}{q_1^3} - \frac{q_3 l_3}{q_1^2} + \frac{q_3^2 l_1}{q_1^3} - \frac{q_4 l_4}{q_1^2} + \frac{q_4^2 l_1}{q_1^3}\right] \tag{F.7}$$

$$U_1(q,m) = \begin{bmatrix} 0 \\ \frac{4}{3}\mu\left(\frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2}\right) \\ \mu\left(\frac{m_3}{q_1} - \frac{q_3 m_1}{q_1^2}\right) \\ \mu\left(\frac{m_4}{q_1} - \frac{q_4 m_1}{q_1^2}\right) \\ \mu\frac{q_3}{q_1}\left(\frac{m_3}{q_1} - \frac{q_3 m_1}{q_1^2}\right) + \mu\frac{q_4}{q_1}\left(\frac{m_4}{q_1} - \frac{q_4 m_1}{q_1^2}\right) + \frac{4}{3}\mu\frac{q_2}{q_1}\left(\frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2}\right) + \\ \frac{k}{c_v}\left(\frac{m_5}{q_1} - \frac{q_5 m_1}{q_1^2} - \frac{q_2 m_2}{q_1^2} + \frac{q_2^2 m_1}{q_1^3} - \frac{q_3 m_3}{q_1^2} + \frac{q_3^2 m_1}{q_1^3} - \frac{q_4 m_4}{q_1^2} + \frac{q_4^2 m_1}{q_1^3}\right) \end{bmatrix} \tag{F.8}$$

$$U_2(q,n) = \begin{bmatrix} 0 \\ -\frac{2}{3}\mu\left(\frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2}\right) \\ \mu\left(\frac{n_2}{q_1} - \frac{q_2 n_1}{q_1^2}\right) \\ 0 \\ \mu\frac{q_3}{q_1}\left(\frac{n_2}{q_1} - \frac{q_2 n_1}{q_1^2}\right) - \frac{2}{3}\mu\frac{q_2}{q_1}\left(\frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2}\right) \end{bmatrix} \tag{F.9}$$

$$U_3(q,l) = \begin{bmatrix} 0 \\ -\frac{2}{3}\mu\left(\frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}\right) \\ 0 \\ \mu\left(\frac{l_2}{q_1} - \frac{q_2 l_1}{q_1^2}\right) \\ \mu\frac{q_4}{q_1}\left(\frac{l_2}{q_1} - \frac{q_2 l_1}{q_1^2}\right) - \frac{2}{3}\mu\frac{q_2}{q_1}\left(\frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}\right) \end{bmatrix} \tag{F.10}$$

$$V_1(q,m) = \begin{bmatrix} 0 \\ \mu\left(\frac{m_3}{q_1} - \frac{q_3 m_1}{q_1^2}\right) \\ -\frac{2}{3}\mu\left(\frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2}\right) \\ 0 \\ \mu\frac{q_2}{q_1}\left(\frac{m_3}{q_1} - \frac{q_3 m_1}{q_1^2}\right) - \frac{2}{3}\mu\frac{q_3}{q_1}\left(\frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2}\right) \end{bmatrix} \tag{F.11}$$

$$V_2(q,n) = \begin{bmatrix} 0 \\ \mu\left(\frac{n_2}{q_1} - \frac{q_2 n_1}{q_1^2}\right) \\ \frac{4}{3}\mu\left(\frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2}\right) \\ \mu\left(\frac{n_4}{q_1} - \frac{q_4 n_1}{q_1^2}\right) \\ \mu\frac{q_2}{q_1}\left(\frac{n_2}{q_1} - \frac{q_2 n_1}{q_1^2}\right) + \mu\frac{q_4}{q_1}\left(\frac{n_4}{q_1} - \frac{q_4 n_1}{q_1^2}\right) + \frac{4}{3}\mu\frac{q_3}{q_1}\left(\frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2}\right) + \\ \frac{k}{c_v}\left(\frac{n_5}{q_1} - \frac{q_5 n_1}{q_1^2} - \frac{q_2 n_2}{q_1^2} + \frac{q_2^2 n_1}{q_1^3} - \frac{q_3 n_3}{q_1^2} + \frac{q_3^2 n_1}{q_1^3} - \frac{q_4 n_4}{q_1^2} + \frac{q_4^2 n_1}{q_1^3}\right) \end{bmatrix} \tag{F.12}$$

$$V_3(q,l) = \begin{bmatrix} 0 \\ 0 \\ -\frac{2}{3}\mu\left(\frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}\right) \\ \mu\left(\frac{l_3}{q_1} - \frac{q_3 l_1}{q_1^2}\right) \\ \mu\frac{q_4}{q_1}\left(\frac{l_3}{q_1} - \frac{q_3 l_1}{q_1^2}\right) - \frac{2}{3}\mu\frac{q_3}{q_1}\left(\frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}\right) \end{bmatrix} \tag{F.13}$$

$$W_1(q,m) = \begin{bmatrix} 0 \\ \mu\left(\frac{m_4}{q_1} - \frac{q_4 m_1}{q_1^2}\right) \\ 0 \\ -\frac{2}{3}\mu\left(\frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2}\right) \\ \mu\frac{q_2}{q_1}\left(\frac{m_4}{q_1} - \frac{q_4 m_1}{q_1^2}\right) - \frac{2}{3}\mu\frac{q_4}{q_1}\left(\frac{m_2}{q_1} - \frac{q_2 m_1}{q_1^2}\right) \end{bmatrix} \tag{F.14}$$

$$W_2(q,n) = \begin{bmatrix} 0 \\ 0 \\ \mu\left(\frac{n_4}{q_1} - \frac{q_4 n_1}{q_1^2}\right) \\ -\frac{2}{3}\mu\left(\frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2}\right) \\ \mu\frac{q_3}{q_1}\left(\frac{n_4}{q_1} - \frac{q_4 n_1}{q_1^2}\right) - \frac{2}{3}\mu\frac{q_3}{q_1}\left(\frac{n_3}{q_1} - \frac{q_3 n_1}{q_1^2}\right) \end{bmatrix} \tag{F.15}$$

$$W_3(q,l) = \begin{bmatrix} 0 \\ \mu\left(\frac{l_2}{q_1} - \frac{q_2 l_1}{q_1^2}\right) \\ \mu\left(\frac{l_3}{q_1} - \frac{q_3 l_1}{q_1^2}\right) \\ \frac{4}{3}\mu\left(\frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}\right) \\ \mu\frac{q_2}{q_1}\left(\frac{l_2}{q_1} - \frac{q_2 l_1}{q_1^2}\right) + \mu\frac{q_3}{q_1}\left(\frac{l_3}{q_1} - \frac{q_3 l_1}{q_1^2}\right) + \frac{4}{3}\mu\frac{q_4}{q_1}\left(\frac{l_4}{q_1} - \frac{q_4 l_1}{q_1^2}\right) + \\ \frac{k}{c_v}\left(\frac{l_5}{q_1} - \frac{q_5 l_1}{q_1^2} - \frac{q_2 l_2}{q_1^2} + \frac{q_2^2 l_1}{q_1^3} - \frac{q_3 l_3}{q_1^2} + \frac{q_3^2 l_1}{q_1^3} - \frac{q_4 l_4}{q_1^2} + \frac{q_4^2 l_1}{q_1^3}\right) \end{bmatrix} \tag{F.16}$$

Mathematica is employed to generate $R = \frac{\partial U_1}{\partial Q_x}$, $R_1 = \frac{\partial U_2}{\partial Q_y}$, $R_2 = \frac{\partial U_3}{\partial Q_z}$, $S = \frac{\partial V_2}{\partial Q_y}$, $S_1 = \frac{\partial V_1}{\partial Q_x}$, $S_2 = \frac{\partial V_3}{\partial Q_z}$, $Y = \frac{\partial W_3}{\partial Q_z}$, $Y_1 = \frac{\partial W_1}{\partial Q_x}$, $Y_2 = \frac{\partial W_2}{\partial Q_y}$ using the above expressions (see *Appendix 9* for the Mathematica sequence of commands)

Appendix G

Mathematica Sequence used to generate the Fortran code for R, R_1, R_2, S, S_1, S_2, Y, Y_1, Y_2 for the Analysis of Navier-Stokes Equations

```
m={m1,m2,m3,m4,m5}

n={n1,n2,n3,n4,n5}

l={l1,l2,l3,l4,l5}

u1={ 0,

      (4 mu/3)(m2/q1-q2 m1/q1^2),

      mu(m3/q1-q3 m1/q1^2),

      mu(m4/q1-q4 m1/q1^2),

      (mu q4/q1)(m4/q1-q4 m1/q1^2)+(mu q3/q1)(m3/q1-q3 m1/q1^2)+(4 mu q2/
      (3 q1))(m2/q1-q2 m1/q1^2)+(k/cp)(m5/q1-q5 m1/q1^2-q2 m2/q1^2+q2^2
      m1/q1^3 -q3 m3/q1^2+q3^2 m1/q1^3-q4 m4/q1^2+q4^2 m1/q1^3)

   }

u2={ 0,

      (-2 mu/3)(n3/q1-q3 n1/q1^2),

      mu(n2/q1-q2 n1/q1^2),

      0,

      (mu q3/q1)(n2/q1-q2 n1/q1^2)-(2 mu q2/(3 q1))(n3/q1-q3 n1/q1^2)

      }
```

```
u3={ 0,
        (-2 mu/3)(l4/q1-q4 l1/q1^2),
        0,
        mu(l2/q1-q2 l1/q1^2),
        (mu q4/q1)(l2/q1-q2 l1/q1^2)-(2 mu q2/(3 q1))(l4/q1-q4 l1/q1^2)
    }
v1={ 0,
        mu(m3/q1-q3 m1/q1^2),
        (-2 mu/3)(m2/q1-q2 m1/q1^2),
        0,
        (mu q2/q1)(m3/q1-q3 m1/q1^2)-(2 mu q3/(3 q1))(m2/q1-q2 m1/q1^2)
    }
v2={ 0,
        mu(n2/q1-q2 n1/q1^2),
        (4 mu/3)(n3/q1-q3 n1/q1^2),
        mu(n4/q1-q4 n1/q1^2),
        (mu q4/q1)(n4/q1-q4 n1/q1^2)+(mu q2/q1)(n2/q1-q2 n1/q1^2)+(4 mu q3/(3
        q1))(n3/q1-q3 n1/q1^2)+(k/cp)(n5/q1-q5 n1/q1^2-q2 n2/q1^2+q2^2
        n1/q1^3 -q3 n3/q1^2+q3^2 n1/q1^3-q4 n4/q1^2+q4^2 n1/q1^3)
    }
```

```
v3={ 0,
     0,
     (-2 mu/3)(l4/q1-q4 l1/q1^2),
     mu(l3/q1-q3 l1/q1^2),
     (mu q4/q1)(l3/q1-q3 l1/q1^2)-(2 mu q3/(3 q1))(l4/q1-q4 l1/q1^2)
   }
w1={ 0,
     mu(m4/q1-q4 m1/q1^2),
     0,
     (-2 mu/3)(m2/q1-q2 m1/q1^2),
     (mu q2/q1)(m4/q1-q4 m1/q1^2)-(2 mu q4/(3 q1))(m2/q1-q2 m1/q1^2)
}
w2={ 0,
     0,
     mu(n4/q1-q4 n1/q1^2),
     (-2 mu/3)(n3/q1-q3 n1/q1^2),
     (mu q3/q1)(n4/q1-q4 n1/q1^2)-(2 mu q4/(3 q1))(n3/q1-q3 n1/q1^2)
   }
w3={ 0,
     mu(l2/q1-q2 l1/q1^2),
```

```
        mu(l3/q1-q3 l1/q1^2),

        (4 mu/3)(l4/q1-q4 l1/q1^2),

        (mu q3/q1)(l3/q1-q3 l1/q1^2)+(mu q2/q1)(l2/q1-q2 l1/q1^2)+(4 mu q4/(3
        q1))(l4/q1-q4 l1/q1^2)+(k/cp)(l5/q1-q5 l1/q1^2-q2 l2/q1^2+q2^2 l1
        /q1^3 -q3 l3/q1^2+q3^2 l1/q1^3-q4 l4/q1^2+q4^2 l1/q1^3)

    }

r =Factor[ Table[ D[ u1[[i]], m[[j]] ],{i,5},{j,5}] ]

r1=Factor[ Table[ D[ u2[[i]], n[[j]] ],{i,5},{j,5}] ]

r2=Factor[ Table[ D[ u3[[i]], l[[j]] ],{i,5},{j,5}] ]

s =Factor[ Table[ D[ v2[[i]], n[[j]] ],{i,5},{j,5}] ]

s1=Factor[ Table[ D[ v1[[i]], m[[j]] ],{i,5},{j,5}] ]

s2=Factor[ Table[ D[ v3[[i]], l[[j]] ],{i,5},{j,5}] ]

y =Factor[ Table[ D[ w3[[i]], l[[j]] ],{i,5},{j,5}] ]

y1=Factor[ Table[ D[ w1[[i]], m[[j]] ],{i,5},{j,5}] ]

y2=Factor[ Table[ D[ w2[[i]], n[[j]] ],{i,5},{j,5}] ]

q1=rho

q2=rho u

q3=rho v

q4=rho w

q5=rho et
```

```
k =mu cp/Pr

r =Factor[r]

r1=Factor[r1]

r2=Factor[r2]

s =Factor[s]

s1=Factor[s1]

s2=Factor[s2]

y =Factor[y]

y1=Factor[y1]

y2=Factor[y2]

Do[

    ToString[

        StringForm[

            "r(‘‘,‘‘)=‘‘",i,j,FortranForm[r[[i,j]]]

        ]

        ] >>>xxx,

    {i,5},{j,5}

  ]

Do[

    ToString[
```

```
                    StringForm[
                        "r1(‘‘,‘‘)=‘‘",i,j,FortranForm[r1[[i,j]]]
                    ]
                    ] >>>xxx,
        {i,5},{j,5}
    ]
Do[
    ToString[
                    StringForm[
                        "r2(‘‘,‘‘)=‘‘",i,j,FortranForm[r2[[i,j]]]
                    ]
                    ] >>>xxx,
        {i,5},{j,5}
    ]
Do[
    ToString[
                    StringForm[
                        "s(‘‘,‘‘)=‘‘",i,j,FortranForm[s[[i,j]]]
                    ]
                    ] >>>xxx,
```

```
        {i,5},{j,5}
    ]
Do[
    ToString[
                StringForm[
                    "s1(‘‘,‘‘)=‘‘",i,j,FortranForm[s1[[i,j]]]
                ]
                ] >>>xxx,
    {i,5},{j,5}
    ]
Do[
    ToString[
                StringForm[
                    "s2(‘‘,‘‘)=‘‘",i,j,FortranForm[s2[[i,j]]]
                ]
                ] >>>xxx,
    {i,5},{j,5}
    ]
Do[
    ToString[
```

```
                    StringForm[
                        "y(‘‘,‘‘)=‘‘",i,j,FortranForm[y[[i,j]]]
                    ]
                    ] >>>xxx,
        {i,5},{j,5}
    ]
    Do[
        ToString[
                    StringForm[
                        "y1(‘‘,‘‘)=‘‘",i,j,FortranForm[y1[[i,j]]]
                    ]
                    ] >>>xxx,
        {i,5},{j,5}
      ]
    Do[
        ToString[
                    StringForm[
                        "y2(‘‘,‘‘)=‘‘",i,j,FortranForm[y2[[i,j]]]
                    ]
                    ] >>>xxx,
```

{i,5},{j,5}

]

Appendix H

Computed Residual Growth Rates For Proteus Test Problems

EULER FLOW OVER A CYLINDER
STANDARD GRID 21 X 51

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
20	0.930795	0.908427	0.917327
30	0.888906	0.895495	0.913527
40	0.832695	0.848400	0.830117
50	0.948464	0.940807	0.934040
60	0.951933	0.945977	0.940652
70	0.947928	0.952122	0.953439
80	0.946352	0.940380	0.967108
90	0.943105	0.932979	0.973934
100	0.919426	0.924721	0.910288
110	0.929923	0.937277	0.898949
120	0.941130	0.939716	0.936658
130	0.951084	0.942771	0.989041
140	0.927299	0.926370	0.938147
150	0.917412	0.925403	0.885518
160	0.936454	0.935097	0.945436
170	0.940514	0.932715	0.964566
180	0.918601	0.923250	0.906974
190	0.931104	0.939515	0.899311
200	0.948791	0.946918	0.971010
Convergence rate for x-momentum equation			
20	0.912634	0.894159	0.908133
30	0.884359	0.904262	0.858055
40	0.952725	0.955742	0.918973
50	0.953946	0.942842	0.971569
60	0.947580	0.946023	0.955972
70	0.937411	0.936821	0.940752
80	0.932982	0.935483	0.929812
90	0.934067	0.933648	0.932057
100	0.934156	0.936524	0.931436
110	0.934570	0.933310	0.931939
120	0.935263	0.933340	0.936514
130	0.934139	0.934774	0.933654
140	0.927418	0.927906	0.933065
150	0.929302	0.928213	0.926448
160	0.934760	0.935187	0.934262
170	0.931980	0.932460	0.934790
180	0.928875	0.932456	0.931675
190	0.937714	0.936477	0.930370
200	0.944295	0.943643	0.954803
Convergence rate for y-momentum equation			
20	0.938592	0.902479	0.986125
30	0.875004	0.888843	0.885963
40	0.946670	0.962121	0.874463
50	0.966574	0.948683	0.993974
60	0.962191	0.952233	0.975283
70	0.949745	0.947821	0.958014
80	0.937294	0.942102	0.939747
90	0.933285	0.935485	0.930090
100	0.934770	0.934075	0.933758
110	0.933651	0.932586	0.936226
120	0.935002	0.931110	0.935112
130	0.935770	0.938447	0.933129
140	0.928539	0.930206	0.929192
150	0.926840	0.925542	0.932591

160	0.933899	0.932563	0.936830
170	0.933109	0.933133	0.929529
180	0.930466	0.930703	0.929403
190	0.938007	0.935502	0.937970
200	0.945751	0.945721	0.959216

EULER FLOW OVER A CYLINDER
TWICE AS COARSE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
20	0.838248	0.838274	0.834294
30	0.956400	0.957698	0.934468
40	0.940676	0.942493	0.922472
50	0.921385	0.922744	0.947969
60	0.913916	0.908285	0.914900
70	0.927979	0.923816	0.924991
80	0.929017	0.917553	0.962759
90	0.917393	0.909659	0.928159
100	0.906117	0.904644	0.909506
110	0.906169	0.903582	0.906276
120	0.914064	0.912727	0.916575
130	0.916153	0.911361	0.919216
140	0.911645	0.906192	0.913590
Convergence rate for x-momentum equation			
20	0.881918	0.874352	0.914724
30	0.942745	0.945055	0.906679
40	0.924141	0.925601	0.946330
50	0.932285	0.928169	0.944494
60	0.939754	0.932477	0.934226
70	0.926889	0.917204	0.936021
80	0.913862	0.910215	0.930996
90	0.903249	0.898748	0.916804
100	0.901098	0.902392	0.905581
110	0.910659	0.914220	0.903627
120	0.915005	0.912825	0.912998
130	0.912763	0.907539	0.917389
140	0.912400	0.904671	0.911914
Convergence rate for y-momentum equation			
20	0.847251	0.859529	0.813734
30	0.974174	0.967034	0.981948
40	0.948546	0.948147	0.951398
50	0.922454	0.921312	0.911818
60	0.914546	0.908356	0.921013
70	0.927277	0.919256	0.955883
80	0.926306	0.921831	0.930860
90	0.913726	0.913014	0.914262
100	0.902813	0.901317	0.914872
110	0.903976	0.901245	0.905973
120	0.914671	0.913816	0.908560
130	0.916619	0.913130	0.916088
140	0.912293	0.906262	0.931719

EULER FLOW OVER A CYLINDER
TWICE AS FINE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
20	0.999132	0.978953	1.08220
30	0.949807	0.950957	0.935938
40	0.993516	0.967842	0.993892
50	0.965277	0.948236	0.967920
60	0.922663	0.938966	0.949610
70	0.895254	0.935504	0.882762
80	0.944288	0.951219	0.877115
90	0.948587	0.938997	0.996460
100	0.981545	0.984612	0.950276
110	0.953007	0.948991	0.968724
120	1.01894	0.993200	1.04987
130	0.968554	0.941746	0.963302
140	0.917311	0.934880	0.952450
150	0.930264	0.958608	0.856320
160	0.937875	0.922657	0.977529
170	0.988990	0.992632	0.982324
180	0.949260	0.948103	0.946159
190	0.979984	0.993645	0.967779
200	0.987924	0.955449	1.01958
210	0.964980	0.966086	0.985982
220	0.945983	0.958321	0.902053
230	0.953936	0.925696	0.984895
240	0.965728	0.974292	0.976299
250	0.956385	0.969232	0.939454
260	0.983930	0.980622	0.983114
270	0.951225	0.941597	0.967900
280	0.974634	0.988044	0.983972
290	0.974434	0.964299	0.932103
300	0.951064	0.934050	1.00579
310	0.966667	0.984036	0.949288
320	0.968314	0.965746	0.961392
330	0.965600	0.961939	0.980489
340	0.960097	0.955314	0.959725
350	0.971197	0.986513	0.965899
360	0.966996	0.955966	0.964819
370	0.960645	0.955007	0.975052
380	0.968411	0.981890	0.953550
390	0.966408	0.955684	0.973859
400	0.962264	0.960850	0.965915
410	0.966227	0.974555	0.962157
Convergence rate for x-momentum equation			
20	1.00222	0.985602	1.07376
30	0.937967	0.952150	0.932630
40	0.976564	0.942363	0.994335
50	0.974174	0.972816	0.961713
60	0.949965	0.956819	0.961858
70	0.917774	0.933238	0.893293
80	0.954746	0.943761	0.895011
90	0.983113	1.00023	0.986011
100	0.973815	0.964595	0.984160
110	0.975347	0.982240	0.981918
120	0.972074	0.950115	0.976508
130	0.974033	0.970537	0.975508
140	0.965748	0.976428	0.963527
150	0.964262	0.943677	0.971384
160	0.967373	0.971914	0.965584

170	0.967101	0.965547	0.965754
180	0.969449	0.979796	0.967948
190	0.965376	0.954738	0.966872
200	0.967492	0.970700	0.965562
210	0.967896	0.971218	0.965768
220	0.965402	0.957409	0.968992
230	0.966001	0.968119	0.964024
240	0.966374	0.964943	0.966868
250	0.966322	0.965928	0.966825
260	0.965648	0.965364	0.965611
270	0.966498	0.969293	0.965733
280	0.965838	0.961857	0.966240
290	0.965780	0.967107	0.966495
300	0.965964	0.966908	0.964766
310	0.965690	0.963240	0.966404
320	0.965745	0.966604	0.966118
330	0.965653	0.965648	0.965224
340	0.965899	0.966253	0.965607
350	0.965503	0.964355	0.966407
360	0.965672	0.966709	0.965274
370	0.965772	0.965454	0.965128
380	0.965534	0.964867	0.966664
390	0.965625	0.966177	0.965094
400	0.965615	0.965412	0.965191
410	0.965636	0.965321	0.966428

Convergence rate for y-momentum equation

20	0.993385	0.989231	1.08074
30	0.946660	0.948145	0.943018
40	0.980815	0.957554	1.01642
50	0.977163	0.977195	0.973673
60	0.934256	0.943058	0.917933
70	0.910370	0.935106	0.875549
80	0.938308	0.926763	0.987414
90	0.994473	0.999260	0.963446
100	0.983342	0.985741	0.961404
110	0.974585	0.974188	0.983625
120	0.971491	0.944565	0.980105
130	0.979403	0.982617	0.971972
140	0.960645	0.971977	0.974614
150	0.961180	0.938926	0.970283
160	0.967978	0.967031	0.972597
170	0.968150	0.970086	0.973454
180	0.969005	0.979111	0.972552
190	0.966869	0.953495	0.970628
200	0.969278	0.974196	0.969674
210	0.967159	0.969126	0.968866
220	0.966760	0.958180	0.967770
230	0.966184	0.966313	0.967558
240	0.966371	0.964897	0.967832
250	0.966576	0.965400	0.967386
260	0.966766	0.966227	0.967039
270	0.966035	0.970740	0.966746
280	0.965916	0.958871	0.966641
290	0.966799	0.969526	0.966346
300	0.965404	0.966793	0.966280
310	0.965826	0.962339	0.966167
320	0.966172	0.966359	0.966009
330	0.965612	0.966416	0.965978
340	0.965808	0.966469	0.965935
350	0.965886	0.963086	0.965857
360	0.965772	0.968160	0.965773
370	0.965529	0.965172	0.965814
380	0.965838	0.964372	0.965702
390	0.965604	0.966577	0.965688

400	0.965566	0.965704	0.965696
410	0.965714	0.964875	0.965622

VISCOUS FLOW OVER A CYLINDER
STANDARD GRID 51 X 51

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
20	0.967640	0.960502	1.00755
30	0.959213	0.943479	1.03628
40	0.919643	0.929059	0.886886
50	0.990995	0.962196	1.01585
60	0.971054	0.961648	0.987030
70	0.988279	0.985485	0.970334
80	0.987387	0.977840	1.00365
90	0.982145	0.979593	0.977281
100	0.981899	0.972972	0.987464
110	0.976589	0.975546	0.976735
120	0.972774	0.965939	0.973696
130	0.968435	0.969022	0.970886
140	0.962888	0.960584	0.962731
150	0.958405	0.960378	0.960289
160	0.952761	0.952681	0.951099
170	0.948241	0.952925	0.945870
180	0.945125	0.949144	0.935203
190	0.944299	0.950688	0.926095
200	0.950171	0.953000	0.945483
210	0.956200	0.953196	0.965326
220	0.960636	0.956454	0.973436
230	0.959405	0.952708	0.966612
240	0.955966	0.950796	0.953588
250	0.952699	0.949536	0.942550
260	0.951081	0.949436	0.948421
270	0.953168	0.954622	0.945376
280	0.957946	0.956586	0.956518
290	0.961462	0.960322	0.969354
300	0.961358	0.960091	0.964340
310	0.958290	0.960178	0.958961
320	0.953999	0.957266	0.957276
330	0.950288	0.955713	0.950040
340	0.949547	0.955925	0.944027
350	0.954123	0.957456	0.941233
Convergence rate for x-momentum equation			
20	0.980322	0.965641	1.08783
30	0.992505	0.977823	0.991679
40	0.978029	0.965791	0.919779
50	0.987489	0.987585	0.991064
60	0.982406	0.970906	0.986907
70	0.984036	0.985564	0.981746
80	0.982885	0.975709	0.983015
90	0.980705	0.977410	0.979493
100	0.978792	0.973928	0.979966
110	0.974396	0.969635	0.975161
120	0.970482	0.966613	0.970490
130	0.965081	0.961011	0.967377
140	0.959950	0.957859	0.959644
150	0.954328	0.953911	0.955582
160	0.948991	0.950294	0.947888
170	0.944699	0.947872	0.943345
180	0.943283	0.946757	0.947882
190	0.947262	0.948225	0.959369
200	0.955556	0.954683	0.958868
210	0.961106	0.956624	0.958528
220	0.961112	0.958070	0.956810

230	0.957704	0.955700	0.953308
240	0.953833	0.953023	0.947772
250	0.951546	0.953300	0.948159
260	0.952554	0.955228	0.962912
270	0.956909	0.956454	0.963201
280	0.960941	0.957836	0.962277
290	0.961592	0.957670	0.960265
300	0.958904	0.955416	0.956569
310	0.954495	0.954871	0.951098
320	0.950107	0.955499	0.945454
330	0.948081	0.954385	0.947084
340	0.951624	0.956732	0.964407
350	0.960668	0.961036	0.964605

Convergence rate for y-momentum equation

20	0.977823	0.973260	1.05152
30	0.991898	0.976226	0.969461
40	0.993988	0.976802	1.00256
50	0.997232	0.990891	1.00881
60	0.989318	0.974185	1.00140
70	0.988187	0.985909	0.996797
80	0.987648	0.979421	0.994527
90	0.985776	0.981824	0.991585
100	0.983033	0.977614	0.986551
110	0.978743	0.974182	0.980053
120	0.974189	0.970189	0.977906
130	0.969098	0.965087	0.971046
140	0.963938	0.961335	0.967696
150	0.958889	0.956190	0.960730
160	0.954375	0.953389	0.952985
170	0.951042	0.952220	0.950530
180	0.949573	0.951393	0.945528
190	0.950532	0.950810	0.949807
200	0.953614	0.952892	0.960892
210	0.956344	0.954421	0.959353
220	0.957282	0.956673	0.957766
230	0.956361	0.955426	0.955305
240	0.954891	0.953920	0.953193
250	0.954060	0.953699	0.951861
260	0.954783	0.955454	0.949106
270	0.956924	0.956915	0.953568
280	0.958933	0.958853	0.965848
290	0.959446	0.959408	0.963189
300	0.958084	0.957674	0.959521
310	0.955599	0.954785	0.955062
320	0.953061	0.953713	0.954356
330	0.951837	0.954087	0.949922
340	0.953122	0.955802	0.946775
350	0.957173	0.958285	0.954036

VISCOUS FLOW OVER A CYLINDER
TWICE AS COARSE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
20	0.942996	0.919789	0.936710
30	1.00508	0.975670	1.03357
40	0.986658	0.965748	1.00901
50	0.964220	0.955625	0.970917
60	0.954498	0.955778	0.952780
70	0.950842	0.948671	0.950095
80	0.943451	0.939874	0.945953
90	0.933722	0.931233	0.936722
100	0.924937	0.923467	0.927251
110	0.917686	0.917801	0.919322
120	0.911532	0.914708	0.910396
130	0.910472	0.917621	0.915336
140	0.925418	0.934058	0.900810
150	0.944203	0.941361	0.937576
160	0.945464	0.941822	0.953143
170	0.937773	0.935502	0.938530
180	0.928485	0.925850	0.926137
190	0.919715	0.920431	0.915835
200	0.914709	0.919719	0.934082
Convergence rate for x-momentum equation			
20	0.984496	0.968344	0.926718
30	0.994231	0.982288	0.996788
40	0.971726	0.965699	0.973688
50	0.963416	0.958422	0.988744
60	0.962593	0.952964	0.994809
70	0.957248	0.947540	0.975079
80	0.945127	0.938430	0.954262
90	0.932495	0.928763	0.938112
100	0.921148	0.918514	0.924836
110	0.908617	0.908671	0.909719
120	0.892111	0.907311	0.882691
130	0.900077	0.926049	0.849242
140	0.973604	0.956899	1.00005
150	0.970912	0.951405	0.998648
160	0.952063	0.941742	0.960194
170	0.937314	0.931718	0.941661
180	0.924248	0.921476	0.926741
190	0.911216	0.914328	0.911345
200	0.898690	0.918380	0.888844
Convergence rate for y-momentum equation			
20	0.966322	0.957079	0.908747
30	1.01022	0.994192	1.02264
40	0.990151	0.976268	1.02085
50	0.963673	0.960467	0.971147
60	0.946989	0.949594	0.942469
70	0.939844	0.942532	0.935044
80	0.933797	0.933610	0.932624
90	0.926748	0.930688	0.923985
100	0.920756	0.925390	0.911148
110	0.918401	0.921604	0.897535
120	0.920957	0.923822	0.913927
130	0.928287	0.925082	0.937308
140	0.935630	0.932259	0.935462
150	0.937617	0.934742	0.932753

160	0.934740	0.935594	0.938725
170	0.930223	0.931990	0.923618
180	0.926440	0.929064	0.928389
190	0.925326	0.926806	0.922451
200	0.928265	0.926636	0.934401

VISCOUS FLOW OVER A CYLINDER
TWICE AS FINE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
20	1.03092	1.02840	1.08824
30	0.963524	0.955435	1.01583
40	0.964685	0.979867	0.950464
50	0.991846	0.965286	0.978138
60	1.00767	1.00976	1.03639
70	0.939623	0.956268	0.949646
80	0.978365	0.968256	0.949601
90	0.971803	0.978629	0.976247
100	1.01021	0.983077	1.04658
110	0.942406	0.936715	0.950845
120	0.969869	0.980935	0.954582
130	1.00366	1.00813	1.00339
140	0.999187	1.00107	0.997750
150	1.00619	0.976264	1.02800
160	0.979493	0.981214	0.993961
170	0.980161	0.985766	0.971720
180	0.984973	0.999673	0.977324
190	1.00687	0.976172	1.02712
200	0.981179	0.965751	0.990477
210	0.970729	0.956358	0.970978
220	0.984996	1.01916	0.987431
230	0.992829	0.997044	0.996190
240	0.987095	0.967280	0.989523
250	0.975727	0.968337	0.982915
260	0.970717	0.974007	0.974956
270	0.981453	1.01676	0.975973
280	0.993642	0.984215	0.988227
290	0.980744	0.959378	0.985158
300	0.960442	0.958545	0.970124
310	0.973285	1.00944	0.970074
320	0.994705	0.995892	0.984578
330	0.987936	0.967612	0.985806
340	0.970707	0.962142	0.977537
350	0.967197	0.980529	0.974130
360	0.982818	1.00129	0.978016
370	0.994249	0.986718	0.985291
380	0.985251	0.965841	0.986231
390	0.968078	0.964391	0.978002
400	0.970808	0.992867	0.973584
410	0.991152	0.997928	0.981831
420	0.991233	0.974679	0.987234
430	0.975395	0.964048	0.980784
440	0.968488	0.977724	0.974822
450	0.980603	0.996067	0.977655
460	0.990259	0.987130	0.983418
470	0.984681	0.970675	0.983794
480	0.973008	0.968666	0.979023
490	0.971707	0.984400	0.975588
500	0.984814	0.994457	0.979255
510	0.989688	0.979944	0.984681
520	0.979501	0.968130	0.982367
530	0.970771	0.974441	0.976736
540	0.977785	0.991067	0.977084
550	0.987844	0.987937	0.982344
560	0.984997	0.974336	0.983573
570	0.975801	0.970701	0.979799
580	0.973394	0.981166	0.976998
590	0.981839	0.990537	0.979038

600	0.987200	0.982604	0.982824
610	0.981600	0.972169	0.982473
620	0.973963	0.973983	0.978454
630	0.976439	0.986232	0.977281
640	0.984930	0.987821	0.980928
650	0.985102	0.977499	0.983048
660	0.977947	0.972499	0.980557
670	0.974693	0.979155	0.977813
680	0.980179	0.987502	0.978887
690	0.985240	0.983577	0.981827
700	0.982287	0.975158	0.982118
710	0.976349	0.974712	0.979447
720	0.976497	0.982880	0.978039
730	0.982581	0.986557	0.980044
740	0.984456	0.979839	0.982206
750	0.979661	0.974447	0.981104

Convergence rate for x-momentum equation

20	0.967969	0.970359	1.01940
30	0.989910	0.979664	0.964463
40	0.988140	0.971235	1.03298
50	1.00381	1.02522	1.01839
60	0.993356	0.974618	0.997676
70	0.995422	0.974112	0.993039
80	0.985924	0.982180	0.991614
90	0.989111	1.00479	0.987959
100	0.995959	0.989417	0.999309
110	0.994878	0.983251	0.998604
120	0.989869	0.983930	0.991678
130	0.990588	0.988372	0.991914
140	0.993077	0.998472	0.993600
150	0.992988	0.994693	0.993631
160	0.990981	0.979716	0.994295
170	0.989468	0.982908	0.992165
180	0.988447	0.991952	0.989129
190	0.989169	0.992354	0.989695
200	0.987940	0.980597	0.990406
210	0.985629	0.978960	0.987706
220	0.983806	0.983359	0.984897
230	0.983176	0.985087	0.983966
240	0.982111	0.981623	0.982988
250	0.980195	0.976391	0.981331
260	0.978247	0.973888	0.979573
270	0.976760	0.978469	0.977401
280	0.976338	0.981940	0.975966
290	0.976024	0.976184	0.975852
300	0.975282	0.971563	0.975413
310	0.974743	0.976220	0.974085
320	0.975774	0.982366	0.973564
330	0.977642	0.980498	0.975057
340	0.978469	0.976499	0.976719
350	0.978684	0.976385	0.977797
360	0.979217	0.980624	0.978796
370	0.980377	0.983988	0.979896
380	0.981036	0.981237	0.981047
390	0.980688	0.976746	0.981709
400	0.979984	0.977996	0.981009
410	0.979870	0.983101	0.980112
420	0.980493	0.982801	0.980205
430	0.980418	0.978359	0.980614
440	0.979696	0.977095	0.980139
450	0.979362	0.980126	0.979409
460	0.979795	0.982604	0.979329
470	0.980292	0.981045	0.979770
480	0.980058	0.977964	0.980124

490	0.979704	0.977982	0.980015
500	0.979694	0.981410	0.979532
510	0.980198	0.982633	0.979735
520	0.980530	0.979716	0.980414
530	0.980140	0.977716	0.980506
540	0.979803	0.979713	0.980018
550	0.980058	0.982172	0.979824
560	0.980464	0.981371	0.980208
570	0.980403	0.978892	0.980486
580	0.980017	0.978506	0.980307
590	0.979874	0.980684	0.979968
600	0.980242	0.982039	0.979966
610	0.980441	0.980414	0.980271
620	0.980287	0.978486	0.980426
630	0.979902	0.979331	0.980179
640	0.980021	0.981487	0.979895
650	0.980355	0.981416	0.980109
660	0.980394	0.979502	0.980390
670	0.980109	0.978754	0.980330
680	0.979942	0.980254	0.980060
690	0.980184	0.981573	0.979999
700	0.980404	0.980632	0.980224
710	0.980262	0.979136	0.980373
720	0.980040	0.979298	0.980247
730	0.980066	0.980867	0.979980
740	0.980214	0.981284	0.980149
750	0.980407	0.979963	0.980287

Convergence rate for y-momentum equation

20	0.969371	0.958650	1.04187
30	0.990651	1.00106	1.02009
40	0.955820	0.942234	0.953465
50	1.02980	1.04329	0.988192
60	0.974655	0.960791	1.02048
70	0.981205	0.976287	0.931220
80	0.985913	0.976884	0.996407
90	1.00687	1.01637	1.00905
100	0.990936	0.983667	1.00397
110	0.987603	0.977898	1.00120
120	0.991746	0.980831	1.00270
130	0.996208	0.993121	1.00229
140	0.996479	1.00279	1.00038
150	0.994403	0.993567	0.999221
160	0.990006	0.974215	0.998002
170	0.991434	0.983897	0.996662
180	0.992314	0.997413	0.995372
190	0.991106	0.993237	0.993559
200	0.986498	0.976003	0.991252
210	0.985452	0.977133	0.989851
220	0.985185	0.985376	0.987803
230	0.983455	0.987059	0.985177
240	0.980509	0.980419	0.982355
250	0.977772	0.971889	0.979680
260	0.976050	0.970863	0.978334
270	0.975406	0.981203	0.975757
280	0.974838	0.984349	0.973022
290	0.971499	0.971056	0.970188
300	0.969405	0.964712	0.968737
310	0.972235	0.977501	0.966757
320	0.975974	0.987003	0.966623
330	0.975677	0.979964	0.968073
340	0.974959	0.972205	0.975461
350	0.977947	0.974324	0.980872
360	0.982178	0.984300	0.983328
370	0.983701	0.988480	0.983565

380	0.981233	0.980138	0.982809
390	0.978550	0.972400	0.981704
400	0.980121	0.978248	0.981102
410	0.982540	0.987724	0.980611
420	0.981254	0.983850	0.979719
430	0.977886	0.974827	0.979017
440	0.977908	0.974585	0.978950
450	0.980347	0.982126	0.979199
460	0.981367	0.985559	0.979208
470	0.979844	0.980503	0.979279
480	0.977989	0.974718	0.979521
490	0.979049	0.977006	0.979819
500	0.981395	0.984410	0.980134
510	0.981590	0.984544	0.980168
520	0.979257	0.977665	0.980076
530	0.978501	0.975233	0.980142
540	0.980445	0.980743	0.980320
550	0.981761	0.984865	0.980310
560	0.980548	0.981435	0.980129
570	0.978913	0.976629	0.980085
580	0.979357	0.977352	0.980121
590	0.980878	0.982445	0.980240
600	0.981402	0.983823	0.980264
610	0.979885	0.979452	0.980034
620	0.978888	0.976381	0.980087
630	0.979991	0.979493	0.980197
640	0.981283	0.983513	0.980260
650	0.980777	0.981991	0.980171
660	0.979352	0.977847	0.980086
670	0.979346	0.977652	0.980154
680	0.980603	0.981280	0.980250
690	0.981147	0.983031	0.980255
700	0.980189	0.980329	0.980107
710	0.979326	0.977569	0.980133
720	0.979787	0.978976	0.980234
730	0.980888	0.982291	0.980253
740	0.980830	0.982067	0.980202
750	0.979742	0.978967	0.980126

FLAT PLATE TURBULENT FLOW (Run 2)
STANDARD GRID 81 X 51

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
4370	1.02830	1.03300	1.01720
4430	1.00725	1.00742	1.00128
4490	1.00237	1.00249	1.00026
4550	1.00061	1.00096	0.999673
4610	0.999756	1.00020	0.998388
4670	0.999288	0.999628	0.999001
4730	0.999009	0.999240	0.998574
4790	0.998841	0.999022	0.998404
4850	0.998742	0.998852	0.998275
4910	0.998708	0.998732	1.00304
4970	0.998672	0.998628	1.00218
5030	0.998701	0.998616	1.00150
5090	0.998700	0.998551	1.00099
5150	0.998729	0.998529	1.00054
5210	0.998719	0.998492	1.00134
5270	0.998748	0.998504	1.00133
Convergence rate for x-momentum equation			
4370	1.02551	1.03100	1.01442
4430	1.00667	1.00712	1.00067
4490	1.00219	1.00244	0.999406
4550	1.00058	1.00104	0.998589
4610	0.999826	1.00036	0.998128
4670	0.999411	0.999783	0.998248
4730	0.999147	0.999400	0.998614
4790	0.998971	0.999151	0.998352
4850	0.998839	0.998951	0.997950
4910	0.998738	0.998778	0.998427
4970	0.998650	0.998639	0.998469
5030	0.998578	0.998527	0.998510
5090	0.998506	0.998410	0.998406
5150	0.998446	0.998286	0.999062
5210	0.998411	0.998184	0.998898
5270	0.998399	0.998143	0.998872
Convergence rate for y-momentum equation			
4370	1.01344	1.00680	1.01858
4430	1.00677	1.00661	1.00924
4490	1.00255	1.00152	0.995087
4550	1.00085	1.00191	1.00112
4610	0.999863	0.999418	0.993879
4670	0.999607	1.00007	1.00009
4730	0.999020	0.998925	0.999311
4790	0.999217	0.999149	0.999286
4850	0.998734	0.998933	1.00034
4910	0.999224	0.998769	0.999259
4970	0.998664	0.998985	0.999265
5030	0.999397	0.998709	0.998877
5090	0.998668	0.998935	0.999290
5150	0.999527	0.998718	0.998833
5210	0.998674	0.998931	0.999394
5270	0.999596	0.998901	0.999611

FLAT PLATE TURBULENT FLOW (Run 2)
TWICE AS COARSE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
10110	1.00908	1.01518	1.00337
10230	1.00023	1.00029	1.00212
10350	0.999197	0.998460	1.00171
10470	0.999141	0.998033	1.00031
10590	0.999250	0.998016	0.999618
10710	0.999411	0.997619	1.00035
10830	0.999567	0.998271	1.00028
10950	0.999683	0.998986	1.00011
Convergence rate for x-momentum equation			
10110	1.01470	1.01596	1.01090
10230	0.999791	1.00032	0.995945
10350	0.998144	0.998193	0.998798
10470	0.997703	0.997663	0.998703
10590	0.998227	0.998568	1.00112
10710	0.999387	0.998415	0.999451
10830	0.999744	0.999165	0.998708
10950	0.999257	0.999332	0.998297
Convergence rate for y-momentum equation			
10110	1.01048	1.00841	1.01054
10230	0.999922	1.00018	0.998185
10350	0.998632	0.998276	0.998192
10470	0.998443	0.997978	0.997075
10590	0.998425	0.998158	0.998689
10710	0.998398	0.998253	1.00014
10830	0.998250	0.998117	0.999044
10950	0.998206	0.998021	0.998120

FLAT PLATE TURBULENT FLOW (Run 2)
TWICE AS FINE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
3880	1.03254	1.03929	1.02393
3930	1.00983	1.01244	1.00281
3980	1.00515	1.00561	0.999985
4030	1.00310	1.00329	0.998854
4080	1.00194	1.00192	0.999367
4130	1.00122	1.00132	1.00013
4180	1.00074	1.00077	1.00008
4230	1.00043	1.00052	0.999543
4280	1.00017	1.00033	0.999585
4330	0.999988	1.00016	0.999379
4380	0.999856	1.00008	0.999172
4430	0.999767	0.999924	0.999038
4480	0.999681	0.999836	0.999193
4530	0.999603	0.999735	0.999095
4580	0.999533	0.999653	0.999052
4630	0.999496	0.999600	0.998986
4680	0.999463	0.999533	0.999037
4730	0.999435	0.999510	0.999139
4780	0.999391	0.999448	0.999122
4830	0.999358	0.999420	0.999144
Convergence rate for x-momentum equation			
3880	1.02683	1.03625	1.01687
3930	1.00911	1.01217	1.00259
3980	1.00481	1.00555	1.00006
4030	1.00288	1.00318	0.999035
4080	1.00182	1.00191	0.999176
4130	1.00113	1.00124	0.999089
4180	1.00070	1.00082	0.998957
4230	1.00039	1.00054	0.998998
4280	1.00018	1.00039	0.998968
4330	1.00002	1.00027	0.998894
4380	0.999905	1.00016	0.999171
4430	0.999808	1.00003	0.999046
4480	0.999741	0.999920	0.998992
4530	0.999671	0.999833	0.999120
4580	0.999615	0.999758	0.999050
4630	0.999573	0.999685	0.999003
4680	0.999546	0.999626	0.998975
4730	0.999500	0.999579	0.999056
4780	0.999468	0.999531	0.999142
4830	0.999454	0.999492	0.999131
Convergence rate for y-momentum equation			
3880	1.00590	1.00994	0.998929
3930	1.00904	1.00460	1.00949
3980	1.00543	1.00429	1.00537
4030	1.00094	1.00202	1.00093
4080	1.00072	1.00147	0.998118
4130	1.00153	1.00124	0.998559
4180	1.00168	1.00130	0.999119
4230	1.00016	1.00079	0.997999
4280	0.999344	1.00008	0.996947
4330	0.999613	0.999954	0.998770
4380	1.00015	0.999797	0.999812
4430	1.00020	1.00032	1.00079

4480	0.999544	0.999799	0.999303
4530	0.999145	0.999305	0.999529
4580	0.999348	0.999469	0.999644
4630	0.999647	0.999464	0.999303
4680	0.999812	1.00013	0.999327
4730	0.999327	0.999106	0.999657
4780	0.999216	0.999144	0.999567
4830	0.999203	0.999132	0.999149

FLAT PLATE TURBULENT FLOW (Run 3)
STANDARD GRID 81 X 51

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
5410	0.890961	0.952515	1.09282
5510	0.996646	0.996347	0.999126
5610	0.998534	0.997202	0.999138
5710	0.998734	0.995862	0.999716
5810	0.999255	0.996012	1.00007
5910	0.999708	0.997504	1.00017
6010	0.999922	0.997638	1.00015
6110	0.999972	0.998706	1.00008
6210	0.999977	0.999035	1.00001
6310	0.999982	0.999455	0.999987
6410	0.999986	0.999919	0.999987
6510	0.999991	0.999808	0.999993
6610	0.999991	0.999558	1.00001
6710	0.999991	0.999372	1.00001
6810	0.999991	0.999276	1.00001
6910	0.999995	0.999202	1.00001
7010	1.00000	0.999157	1.00002
7110	0.999995	0.999137	1.00001
7210	1.00000	0.999150	1.00001
7310	0.999995	0.999182	1.00001
7410	1.00000	0.999238	1.00001
7510	1.00000	0.999308	1.00001
7610	1.00000	0.999388	1.00001
7710	1.00000	0.999457	1.00000
7810	0.999995	0.999528	1.00001
7910	1.00000	0.999594	1.00000
8010	1.00000	0.999653	1.00000
8110	1.00000	0.999700	1.00000
8210	1.00000	0.999746	1.00000
8310	1.00000	0.999783	1.00001
8410	1.00000	0.999817	1.00000
8510	1.00000	0.999852	1.00000
8610	1.00000	0.999870	1.00000
8710	1.00000	0.999895	1.00000
8810	1.00000	0.999914	1.00000
8910	1.00000	0.999926	1.00000
9010	1.00000	0.999939	1.00000
9110	1.00000	0.999946	1.00000
9210	1.00000	0.999959	1.00000
9310	1.00000	0.999966	1.00000
9410	1.00000	0.999973	1.00000
9510	1.00000	0.999979	1.00000
9610	0.999995	0.999979	1.00000
9710	1.00000	0.999986	1.00000
9810	1.00000	0.999986	1.00000
9910	1.00000	0.999993	1.00000
10010	1.00000	0.999986	1.00000
10110	1.00000	0.999993	1.00000
10210	1.00000	1.00000	1.00000
10310	1.00000	0.999993	1.00000
10410	1.00000	1.00000	1.00000
10510	1.00000	0.999993	1.00000
10610	1.00000	1.00000	1.00000
10710	1.00000	1.00000	1.00000
10810	1.00000	0.999993	1.00000
10910	1.00000	1.00000	1.00000
11010	1.00000	1.00000	1.00000
11110	1.00000	1.00000	1.00000

11210	1.00000	1.00000	1.00000
11310	1.00000	1.00000	1.00000
11410	1.00000	1.00000	1.00000
11510	1.00000	1.00000	1.00000
11610	1.00000	1.00000	1.00000
11710	1.00000	1.00000	1.00000
11810	1.00000	1.00000	1.00000
11910	1.00000	1.00000	1.00000
12010	1.00000	1.00000	1.00000
12110	1.00000	1.00000	1.00000
12210	1.00000	1.00000	1.00000
12310	1.00000	1.00000	1.00000
12410	1.00000	1.00000	1.00000
12510	1.00000	1.00000	1.00000
12610	1.00000	1.00000	1.00000
12710	1.00000	1.00000	1.00000
12810	1.00000	1.00000	1.00000
12910	1.00000	1.00000	1.00000
13010	1.00000	1.00000	1.00000
13110	1.00000	1.00000	1.00000
13210	1.00000	1.00000	1.00000
13310	1.00000	1.00000	1.00000
13410	1.00000	1.00000	1.00000
13510	1.00000	1.00000	1.00000
13610	1.00000	1.00000	1.00000
13710	1.00000	1.00000	1.00000
13810	1.00000	1.00000	1.00000
13910	1.00000	1.00000	1.00000
14010	1.00000	1.00000	1.00000
14110	1.00000	1.00000	1.00000
14210	1.00000	1.00000	1.00000
14310	1.00000	1.00000	1.00000
14410	1.00000	1.00000	1.00000
14510	1.00000	1.00000	1.00000
14610	1.00000	1.00000	1.00000
14710	1.00000	1.00000	1.00000
14810	1.00000	1.00000	1.00000
14910	1.00000	1.00000	1.00000
15010	1.00000	1.00000	1.00000
15110	1.00000	1.00000	1.00000
15210	1.00000	1.00000	1.00000
15310	1.00000	1.00000	1.00000

Convergence rate for x-momentum equation

5410	0.937167	0.995221	1.10922
5510	0.999017	0.998451	1.00021
5610	0.995244	0.996135	0.993135
5710	0.992674	0.993272	0.994610
5810	0.995606	0.994916	0.997059
5910	0.997944	0.997881	0.999893
6010	0.997718	0.997132	1.00107
6110	0.997998	0.998474	1.00067
6210	0.998247	0.998559	1.00023
6310	0.998676	0.998712	0.999943
6410	0.999196	0.999232	0.999857
6510	0.999323	0.999379	0.999878
6610	0.999201	0.999201	0.999931
6710	0.999063	0.998963	0.999977
6810	0.998985	0.998839	0.999995
6910	0.998941	0.998713	1.00001
7010	0.998913	0.998580	1.00002
7110	0.998906	0.998468	1.00003
7210	0.998941	0.998382	1.00004
7310	0.999018	0.998317	1.00004
7410	0.999134	0.998280	1.00004

7510	0.999267	0.998265	1.00004
7610	0.999403	0.998272	1.00003
7710	0.999529	0.998283	1.00002
7810	0.999640	0.998307	1.00002
7910	0.999728	0.998338	1.00001
8010	0.999798	0.998363	1.00001
8110	0.999851	0.998410	1.00001
8210	0.999891	0.998442	1.00001
8310	0.999924	0.998492	1.00000
8410	0.999945	0.998549	1.00000
8510	0.999960	0.998620	1.00000
8610	0.999972	0.998692	1.00000
8710	0.999981	0.998778	1.00000
8810	0.999988	0.998867	1.00000
8910	0.999991	0.998958	1.00000
9010	0.999991	0.999048	1.00000
9110	0.999997	0.999147	1.00000
9210	0.999997	0.999235	1.00000
9310	1.00000	0.999328	1.00000
9410	0.999997	0.999413	1.00000
9510	1.00000	0.999491	1.00000
9610	1.00000	0.999564	1.00000
9710	0.999997	0.999632	1.00000
9810	1.00000	0.999688	1.00000
9910	1.00000	0.999742	1.00000
10010	1.00000	0.999787	1.00000
10110	1.00000	0.999822	1.00000
10210	1.00000	0.999859	1.00000
10310	1.00000	0.999884	1.00000
10410	1.00000	0.999905	1.00000
10510	1.00000	0.999922	1.00000
10610	1.00000	0.999940	1.00000
10710	1.00000	0.999949	1.00000
10810	1.00000	0.999963	1.00000
10910	1.00000	0.999967	1.00000
11010	1.00000	0.999977	1.00000
11110	1.00000	0.999981	1.00000
11210	1.00000	0.999986	1.00000
11310	1.00000	0.999986	1.00000
11410	1.00000	0.999991	1.00000
11510	1.00000	0.999995	1.00000
11610	1.00000	0.999995	1.00000
11710	1.00000	0.999995	1.00000
11810	1.00000	0.999995	1.00000
11910	1.00000	0.999995	1.00000
12010	1.00000	1.00000	1.00000
12110	1.00000	1.00000	1.00000
12210	1.00000	1.00000	1.00000
12310	1.00000	0.999995	1.00000
12410	1.00000	1.00000	1.00000
12510	1.00000	1.00000	1.00000
12610	1.00000	1.00000	1.00000
12710	1.00000	1.00000	1.00000
12810	1.00000	1.00000	1.00000
12910	1.00000	1.00000	1.00000
13010	1.00000	1.00000	1.00000
13110	1.00000	1.00000	1.00000
13210	1.00000	1.00000	1.00000
13310	1.00000	1.00000	1.00000
13410	1.00000	1.00000	1.00000
13510	1.00000	1.00000	1.00000
13610	1.00000	1.00000	1.00000
13710	1.00000	1.00000	1.00000
13810	1.00000	1.00000	1.00000
13910	1.00000	1.00000	1.00000
14010	1.00000	1.00000	1.00000

14110	1.00000	1.00000	1.00000
14210	1.00000	1.00000	1.00000
14310	1.00000	1.00000	1.00000
14410	1.00000	1.00000	1.00000
14510	1.00000	1.00000	1.00000
14610	1.00000	1.00000	1.00000
14710	1.00000	1.00000	1.00000
14810	1.00000	1.00000	1.00000
14910	1.00000	1.00000	1.00000
15010	1.00000	1.00000	1.00000
15110	1.00000	1.00000	1.00000
15210	1.00000	1.00000	1.00000
15310	1.00000	1.00000	1.00000

Convergence rate for y-momentum equation

5410	0.880516	0.970512	1.09744
5510	0.995021	0.994670	0.997906
5610	0.996398	0.995957	0.998669
5710	0.998050	0.998765	0.997949
5810	0.997905	0.998003	0.999308
5910	0.997865	0.997899	1.00001
6010	0.998200	0.998170	1.00020
6110	0.998507	0.998239	1.00030
6210	0.998713	0.998094	1.00027
6310	0.998957	0.998116	1.00016
6410	0.999200	0.998425	1.00004
6510	0.999339	0.998736	0.999970
6610	0.999403	0.998882	0.999957
6710	0.999464	0.998812	0.999964
6810	0.999556	0.998827	0.999969
6910	0.999648	0.998950	0.999963
7010	0.999718	0.999094	0.999957
7110	0.999759	0.999214	0.999956
7210	0.999803	0.999166	0.999961
7310	0.999828	0.999102	0.999970
7410	0.999861	0.999013	0.999979
7510	0.999882	0.998966	0.999989
7610	0.999910	0.998940	0.999993
7710	0.999932	0.998945	0.999998
7810	0.999947	0.998962	1.00000
7910	0.999962	0.998999	1.00000
8010	0.999969	0.999046	1.00000
8110	0.999985	0.999088	1.00000
8210	0.999985	0.999142	1.00000
8310	0.999985	0.999188	1.00000
8410	0.999992	0.999253	1.00000
8510	0.999992	0.999304	1.00000
8610	0.999992	0.999371	1.00000
8710	1.00000	0.999430	1.00000
8810	1.00000	0.999487	1.00000
8910	0.999992	0.999545	1.00000
9010	1.00000	0.999605	1.00000
9110	1.00000	0.999654	1.00000
9210	1.00000	0.999708	1.00000
9310	1.00000	0.999749	1.00000
9410	1.00000	0.999785	1.00000
9510	1.00000	0.999817	1.00000
9610	1.00000	0.999847	1.00000
9710	1.00000	0.999871	1.00000
9810	1.00000	0.999891	1.00000
9910	1.00000	0.999910	1.00000
10010	1.00000	0.999924	1.00000
10110	1.00000	0.999937	1.00000
10210	1.00000	0.999948	1.00000
10310	1.00000	0.999956	1.00000

10410	1.00000	0.999964	1.00000
10510	1.00000	0.999970	1.00000
10610	1.00000	0.999976	1.00000
10710	1.00000	0.999980	1.00000
10810	1.00000	0.999983	1.00000
10910	1.00000	0.999987	1.00000
11010	1.00000	0.999989	1.00000
11110	1.00000	0.999991	1.00000
11210	1.00000	0.999993	1.00000
11310	1.00000	0.999993	1.00000
11410	1.00000	0.999996	1.00000
11510	1.00000	0.999997	1.00000
11610	1.00000	0.999997	1.00000
11710	1.00000	0.999998	1.00000
11810	1.00000	0.999999	1.00000
11910	1.00000	0.999998	1.00000
12010	1.00000	0.999999	1.00000
12110	1.00000	0.999999	1.00000
12210	1.00000	1.00000	1.00000
12310	1.00000	0.999999	1.00000
12410	1.00000	1.00000	1.00000
12510	1.00000	1.00000	1.00000
12610	1.00000	0.999999	1.00000
12710	1.00000	1.00000	1.00000
12810	1.00000	1.00000	1.00000
12910	1.00000	1.00000	1.00000
13010	1.00000	1.00000	1.00000
13110	1.00000	1.00000	1.00000
13210	1.00000	1.00000	1.00000
13310	1.00000	1.00000	1.00000
13410	1.00000	1.00000	1.00000
13510	1.00000	1.00000	1.00000
13610	1.00000	1.00000	1.00000
13710	1.00000	1.00000	1.00000
13810	1.00000	1.00000	1.00000
13910	1.00000	1.00000	1.00000
14010	1.00000	1.00000	1.00000
14110	1.00000	1.00000	1.00000
14210	1.00000	1.00000	1.00000
14310	1.00000	1.00000	1.00000
14410	1.00000	1.00000	1.00000
14510	1.00000	1.00000	1.00000
14610	1.00000	1.00000	1.00000
14710	1.00000	0.999999	1.00000
14810	1.00000	1.00000	1.00000
14910	1.00000	1.00000	1.00000
15010	1.00000	1.00000	1.00000
15110	1.00000	1.00000	1.00000
15210	1.00000	1.00000	1.00000
15310	1.00000	1.00000	1.00000

FLAT PLATE TURBULENT FLOW (Run 3)
TWICE AS COARSE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
11100	0.999433	0.994436	1.00007
11210	1.00000	0.999461	0.999880
11320	1.00005	1.00016	1.00004
11430	0.999853	0.998951	0.999967
11540	0.999784	0.998639	0.999886
11650	0.999797	0.998779	0.999875
11760	0.999859	0.999376	0.999900
11870	0.999932	0.999836	0.999943
11980	0.999981	0.999657	0.999973
12090	1.00000	0.999513	0.999985
12200	1.00001	0.999415	0.999983
12310	1.00001	0.999406	0.999975
12420	1.00000	0.999458	0.999975
12530	0.999994	0.999537	0.999972
12640	0.999992	0.999622	0.999975
12750	0.999990	0.999711	0.999980
12860	0.999992	0.999779	0.999985
12970	0.999992	0.999837	0.999987
13080	0.999994	0.999889	0.999992
13190	0.999996	0.999920	0.999995
13300	0.999998	0.999952	0.999997
13410	1.00000	0.999968	0.999997
13520	0.999998	0.999984	1.00000
13630	1.00000	0.999984	1.00000
13740	1.00000	0.999992	1.00000
13850	1.00000	1.00000	0.999997
13960	1.00000	1.00000	1.00000
14070	1.00000	0.999992	1.00000
14180	1.00000	1.00000	1.00000
14290	1.00000	1.00000	1.00000
14400	1.00000	1.00000	1.00000
14510	1.00000	1.00000	1.00000
14620	1.00000	1.00000	1.00000
14730	1.00000	1,00000	1.00000
14840	1.00000	1.00000	1.00000
14950	1.00000	1.00000	1.00000
15060	1.00000	1.00000	1.00000
15170	1.00000	1.00000	1.00000
15280	1.00000	1.00000	1.00000
15390	1.00000	1.00000	1.00000
15500	1.00000	1.00000	1.00000
15610	1.00000	1.00000	1.00000
15720	1.00000	1.00000	1.00000
15830	1.00000	1.00000	1.00000
15940	1.00000	1.00000	1.00000
16050	1.00000	1.00000	1.00000
16160	1.00000	1.00000	1.00000
16270	1.00000	1.00000	1.00000
16380	1.00000	1.00000	1.00000
16490	1.00000	1.00000	1.00000
16600	1.00000	1.00000	1.00000
16710	1.00000	1.00000	1.00000
16820	1.00000	1.00000	1.00000
16930	1.00000	1.00000	1.00000
17040	1.00000	1.00000	1.00000
17150	1.00000	1.00000	1.00000
17260	1.00000	1.00000	1.00000
17370	1.00000	1.00000	1.00000

17480	1.00000	1.00000	1.00000
17590	1.00000	1.00000	1.00000
17700	1.00000	1.00000	1.00000
17810	1.00000	1.00000	1.00000
17920	1.00000	1.00000	1.00000
18030	1.00000	1.00000	1.00000
18140	1.00000	1.00000	1.00000
18250	1.00000	1.00000	1.00000
18360	1.00000	1.00000	1.00000
18470	1.00000	1.00000	1.00000
18580	1.00000	1.00000	1.00000
18690	1.00000	1.00000	1.00000
18800	1.00000	1.00000	1.00000
18910	1.00000	1.00000	1.00000
19020	1.00000	1.00000	1.00000
19130	1.00000	1.00000	1.00000
19240	1.00000	1.00000	1.00000
19350	1.00000	1.00000	1.00000
19460	1.00000	1.00000	1.00000
19570	1.00000	1.00000	1.00000
19680	1.00000	1.00000	1.00000
19790	1.00000	1.00000	1.00000
19900	1.00000	1.00000	1.00000
20010	1.00000	1.00000	1.00000
20120	1.00000	1.00000	1.00000
20230	1.00000	1.00000	1.00000
20340	1.00000	1.00000	1.00000
20450	1.00000	1.00000	1.00000
20560	1.00000	1.00000	1.00000
20670	1.00000	1.00000	1.00000
20780	1.00000	1.00000	1.00000
20890	1.00000	1.00000	1.00000
21000	1.00000	1.00000	1.00000

Convergence rate for x-momentum equation

11100	0.988565	0.988862	0.991458
11210	0.999582	0.997893	1.00072
11320	1.00026	0.999739	0.999972
11430	0.998963	0.998567	0.998383
11540	0.998009	0.997947	0.998051
11650	0.997707	0.997525	0.999199
11760	0.997976	0.998455	0.999516
11870	0.998249	0.999431	0.999588
11980	0.998296	0.999088	0.999776
12090	0.998252	0.998673	0.999918
12200	0.998281	0.998231	0.999997
12310	0.998458	0.997851	1.00003
12420	0.998777	0.997591	1.00005
12530	0.999151	0.997441	1.00004
12640	0.999470	0.997361	1.00003
12750	0.999694	0.997359	1.00002
12860	0.999833	0.997407	1.00001
12970	0.999914	0.997551	1.00001
13080	0.999956	0.997811	1.00000
13190	0.999981	0.998180	1.00000
13300	0.999990	0.998591	1.00000
13410	0.999994	0.998966	1.00000
13520	0.999998	0.999276	1.00000
13630	1.00000	0.999494	1.00000
13740	1.00000	0.999679	1.00000
13850	1.00000	0.999802	1.00000
13960	1.00000	0.999882	1.00000
14070	1.00000	0.999932	1.00000
14180	1.00000	0.999966	1.00000
14290	1.00000	0.999966	1.00000

14400	1.00000	0.999991	1.00000
14510	1.00000	0.999991	1.00000
14620	1.00000	0.999991	1.00000
14730	1.00000	1.00000	1.00000
14840	1.00000	0.999991	1.00000
14950	1.00000	1.00000	1.00000
15060	1.00000	1.00000	1.00000
15170	1.00000	1.00000	1.00000
15280	1.00000	1.00000	1.00000
15390	1.00000	1.00000	1.00000
15500	1.00000	1.00000	1.00000
15610	1.00000	1.00000	1.00000
15720	1.00000	1.00000	1.00000
15830	1.00000	1.00000	1.00000
15940	1.00000	1.00000	1.00000
16050	1.00000	1.00000	1.00000
16160	1.00000	1.00000	1.00000
16270	1.00000	1.00000	1.00000
16380	1.00000	1.00000	1.00000
16490	1.00000	1.00000	1.00000
16600	1.00000	1.00000	1.00000
16710	1.00000	1.00000	1.00000
16820	1.00000	1.00000	1.00000
16930	1.00000	1.00000	1.00000
17040	1.00000	1.00000	1.00000
17150	1.00000	1.00000	1.00000
17260	1.00000	1.00000	1.00000
17370	1.00000	1.00000	1.00000
17480	1.00000	1.00000	1.00000
17590	1.00000	1.00000	1.00000
17700	1.00000	1.00000	1.00000
17810	1.00000	1.00000	1.00000
17920	1.00000	1.00000	1.00000
18030	1.00000	1.00000	1.00000
18140	1.00000	1.00000	1.00000
18250	1.00000	1.00000	1.00000
18360	1.00000	1.00000	1.00000
18470	1.00000	1.00000	1.00000
18580	1.00000	1.00000	1.00000
18690	1.00000	1.00000	1.00000
18800	1.00000	1.00000	1.00000
18910	1.00000	1.00000	1.00000
19020	1.00000	1.00000	1.00000
19130	1.00000	1.00000	1.00000
19240	1.00000	1.00000	1.00000
19350	1.00000	1.00000	1.00000
19460	1.00000	1.00000	1.00000
19570	1.00000	1.00000	1.00000
19680	1.00000	1.00000	1.00000
19790	1.00000	1.00000	1.00000
19900	1.00000	1.00000	1.00000
20010	1.00000	1.00000	1.00000
20120	1.00000	1.00000	1.00000
20230	1.00000	1.00000	1.00000
20340	1.00000	1.00000	1.00000
20450	1.00000	1.00000	1.00000
20560	1.00000	1.00000	1.00000
20670	1.00000	1.00000	1.00000
20780	1.00000	1.00000	1.00000
20890	1.00000	1.00000	1.00000
21000	1.00000	1.00000	1.00000

Convergence rate for y-momentum equation

11100	0.992134	0.988640	0.992440
11210	1.00052	1.00054	1.00052

11320	0.998892	1.00012	0.997148
11430	0.998036	0.998688	0.997147
11540	0.997753	0.997458	0.997335
11650	0.998279	0.997465	0.998207
11760	0.998825	0.998375	0.998232
11870	0.998773	0.999329	0.999088
11980	0.998510	0.998843	1.00007
12090	0.998363	0.998185	0.999933
12200	0.998466	0.997747	0.999854
12310	0.998790	0.997765	0.999806
12420	0.999179	0.997952	0.999804
12530	0.999465	0.998267	0.999825
12640	0.999650	0.998568	0.999863
12750	0.999770	0.998745	0.999898
12860	0.999852	0.998851	0.999930
12970	0.999911	0.998955	0.999955
13080	0.999948	0.999057	0.999974
13190	0.999970	0.999182	0.999988
13300	0.999983	0.999337	0.999998
13410	0.999991	0.999489	1.00000
13520	0.999995	0.999625	1.00000
13630	0.999998	0.999740	1.00000
13740	0.999998	0.999820	1.00000
13850	1.00000	0.999883	1.00000
13960	0.999999	0.999925	1.00000
14070	1.00000	0.999954	1.00000
14180	1.00000	0.999968	1.00000
14290	1.00000	0.999982	1.00000
14400	1.00000	0.999988	1.00000
14510	1.00000	0.999991	1.00000
14620	1.00000	0.999997	1.00000
14730	1.00000	0.999997	1.00000
14840	1.00000	1.00000	1.00000
14950	1.00000	0.999997	1.00000
15060	1.00000	1.00000	1.00000
15170	1.00000	1.00000	1.00000
15280	1.00000	1.00000	1.00000
15390	1.00000	1.00000	1.00000
15500	1.00000	1.00000	1.00000
15610	1.00000	1.00000	1.00000
15720	1.00000	1.00000	1.00000
15830	1.00000	1.00000	1.00000
15940	1.00000	1.00000	1.00000
16050	1.00000	1.00000	1.00000
16160	1.00000	1.00000	1.00000
16270	1.00000	1.00000	1.00000
16380	1.00000	1.00000	1.00000
16490	1.00000	1.00000	1.00000
16600	1.00000	1.00000	1.00000
16710	1.00000	1.00000	1.00000
16820	1.00000	1.00000	1.00000
16930	1.00000	1.00000	1.00000
17040	1.00000	1.00000	1.00000
17150	1.00000	1.00000	1.00000
17260	1.00000	1.00000	1.00000
17370	1.00000	1.00000	1.00000
17480	1.00000	1.00000	1.00000
17590	1.00000	1.00000	1.00000
17700	1.00000	1.00000	1.00000
17810	1.00000	1.00000	1.00000
17920	1.00000	1.00000	1.00000
18030	1.00000	1.00000	1.00000
18140	1.00000	1.00000	1.00000
18250	1.00000	1.00000	1.00000
18360	1.00000	1.00000	1.00000
18470	1.00000	1.00000	1.00000

18580	1.00000	1.00000	1.00000
18690	1.00000	1.00000	1.00000
18800	1.00000	1.00000	1.00000
18910	1.00000	1.00000	1.00000
19020	1.00000	1.00000	1.00000
19130	1.00000	1.00000	1.00000
19240	1.00000	1.00000	1.00000
19350	1.00000	1.00000	1.00000
19460	1.00000	1.00000	1.00000
19570	1.00000	1.00000	1.00000
19680	1.00000	1.00000	1.00000
19790	1.00000	1.00000	1.00000
19900	1.00000	1.00000	1.00000
20010	1.00000	1.00000	1.00000
20120	1.00000	1.00000	1.00000
20230	1.00000	1.00000	1.00000
20340	1.00000	1.00000	1.00000
20450	1.00000	1.00000	1.00000
20560	1.00000	1.00000	1.00000
20670	1.00000	1.00000	1.00000
20780	1.00000	1.00000	1.00000
20890	1.00000	1.00000	1.00000
21000	1.00000	1.00000	1.00000

FLAT PLATE TURBULENT FLOW (Run 3)
TWICE AS FINE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
4920	0.992615	0.990764	0.996025
5010	0.996870	0.996612	0.999596
5100	0.997007	0.997108	1.00373
5190	0.996911	0.996655	1.00076
5280	0.997087	0.996937	0.998998
5370	0.997584	0.998375	0.998104
5460	0.998085	0.998530	1.00037
5550	0.998423	0.998385	1.00032
5640	0.998628	0.998233	0.999916
5730	0.998750	0.998105	0.999804
5820	0.998816	0.997885	0.999813
5910	0.998830	0.998209	0.999835
6000	0.998811	0.999006	0.999833
6090	0.998784	0.999237	0.999806
6180	0.998799	0.999049	0.999778
6270	0.998879	0.998879	0.999760
6360	0.999040	0.998860	0.999771
6450	0.999222	0.998995	0.999798
6540	0.999418	0.999196	0.999837
6630	0.999571	0.999370	0.999876
6720	0.999702	0.999527	0.999911
6810	0.999794	0.999657	0.999931
6900	0.999849	0.999743	0.999939
6990	0.999890	0.999786	0.999941
7080	0.999897	0.999792	0.999936
7170	0.999896	0.999770	0.999925
7260	0.999887	0.999733	0.999916
7350	0.999876	0.999689	0.999950
7440	0.999848	0.999646	1.00008
7530	0.999846	0.999604	1.00008
7620	0.999825	0.999559	1.00009
7710	0.999823	0.999522	1.00009
7800	0.999829	0.999493	1.00009
7890	0.999817	0.999460	1.00009
7980	0.999824	0.999433	1.00009
8070	0.999821	0.999405	1.00008
8160	0.999828	0.999381	1.00008
8250	0.999836	0.999354	1.00008
8340	0.999844	0.999332	1.00007
8430	0.999852	0.999311	1.00007
8520	0.999850	0.999286	1.00007
8610	0.999870	0.999266	1.00007
8700	0.999879	0.999251	1.00006
8790	0.999887	0.999235	1.00006
8880	0.999898	0.999221	1.00005
8970	0.999909	0.999212	1.00005
9060	0.999919	0.999200	1.00004
9150	0.999928	0.999196	1.00004
9240	0.999938	0.999189	1.00004
9330	0.999944	0.999193	1.00003
9420	0.999952	0.999186	1.00003
9510	0.999958	0.999191	1.00003
9600	0.999964	0.999193	1.00002
9690	0.999969	0.999190	1.00002
9780	0.999973	0.999202	1.00002
9870	0.999978	0.999209	1.00002
9960	0.999980	0.999214	1.00002
10050	0.999982	0.999224	1.00001

10140	0.999986	0.999233	1.00001
10230	0.999987	0.999241	1.00001
10320	0.999989	0.999254	1.00001
10410	0.999990	0.999267	1.00001
10500	0.999992	0.999279	1.00001
10590	0.999993	0.999294	1.00001
10680	0.999994	0.999307	1.00001
10770	0.999995	0.999326	1.00001
10860	0.999995	0.999340	1.00000
10950	0.999996	0.999362	1.00000
11040	0.999996	0.999379	1.00000
11130	0.999998	0.999398	1.00000
11220	0.999996	0.999420	1.00000
11310	0.999999	0.999441	1.00000
11400	0.999998	0.999464	1.00000
11490	0.999999	0.999488	1.00000
11580	0.999998	0.999510	1.00000
11670	0.999999	0.999533	1.00000
11760	0.999999	0.999559	1.00000
11850	0.999999	0.999581	1.00000
11940	1.00000	0.999602	1.00000
12030	0.999999	0.999630	1.00000
12120	1.00000	0.999649	1.00000
12210	0.999999	0.999674	1.00000
12300	1.00000	0.999698	1.00000
12390	0.999999	0.999718	1.00000
12480	1.00000	0.999741	1.00000
12570	1.00000	0.999755	1.00000
12660	0.999999	0.999780	1.00000
12750	1.00000	0.999797	1.00000
12840	1.00000	0.999809	1.00000
12930	1.00000	0.999827	1.00000
13020	0.999999	0.999841	1.00000
13110	1.00000	0.999861	1.00000
13200	1.00000	0.999865	1.00000
13290	1.00000	0.999881	1.00000
13380	1.00000	0.999897	1.00000
13470	1.00000	0.999902	1.00000
13560	0.999999	0.999912	1.00000
13650	1.00000	0.999918	1.00000
13740	1.00000	0.999929	1.00000

Convergence rate for x-momentum equation

4920	0.996795	0.996715	0.997434
5010	0.996503	0.996092	0.998291
5100	0.996914	0.997513	0.998283
5190	0.998427	0.998024	0.997560
5280	0.999573	0.998537	0.998179
5370	0.999344	0.999320	0.999938
5460	0.998516	0.998748	0.998200
5550	0.997667	0.998060	0.997012
5640	0.997009	0.997464	0.996133
5730	0.996761	0.997005	0.999191
5820	0.997190	0.996726	0.999066
5910	0.998140	0.997392	0.998916
6000	0.998879	0.999238	0.999104
6090	0.999077	0.999572	0.999246
6180	0.998981	0.999218	0.999066
6270	0.998868	0.998913	0.998906
6360	0.998860	0.998826	0.998902
6450	0.998953	0.999020	0.998969
6540	0.999088	0.999316	0.998857
6630	0.999206	0.999395	0.998756
6720	0.999301	0.999431	0.998641
6810	0.999380	0.999466	0.998662

6900	0.999461	0.999499	0.998693
6990	0.999535	0.999530	0.998589
7080	0.999599	0.999551	0.999579
7170	0.999635	0.999566	0.999898
7260	0.999639	0.999569	0.999886
7350	0.999627	0.999572	0.999881
7440	0.999597	0.999555	0.999877
7530	0.999559	0.999537	0.999877
7620	0.999530	0.999511	0.999876
7710	0.999497	0.999482	0.999876
7800	0.999473	0.999456	1.00005
7890	0.999447	0.999429	1.00011
7980	0.999422	0.999398	1.00012
8070	0.999400	0.999378	1.00012
8160	0.999382	0.999342	1.00012
8250	0.999362	0.999326	1.00012
8340	0.999335	0.999292	1.00012
8430	0.999317	0.999283	1.00012
8520	0.999293	0.999244	1.00012
8610	0.999280	0.999233	1.00012
8700	0.999253	0.999212	1.00011
8790	0.999240	0.999192	1.00011
8880	0.999227	0.999177	1.00011
8970	0.999216	0.999161	1.00010
9060	0.999207	0.999147	1.00010
9150	0.999200	0.999134	1.00009
9240	0.999187	0.999123	1.00008
9330	0.999186	0.999112	1.00008
9420	0.999184	0.999102	1.00007
9510	0.999183	0.999092	1.00006
9600	0.999182	0.999088	1.00006
9690	0.999185	0.999076	1.00005
9780	0.999186	0.999071	1.00005
9870	0.999191	0.999065	1.00005
9960	0.999197	0.999057	1.00003
10050	0.999206	0.999051	1.00004
10140	0.999212	0.999046	1.00002
10230	0.999224	0.999038	1.00003
10320	0.999235	0.999031	1.00002
10410	0.999250	0.999031	1.00003
10500	0.999268	0.999020	1.00001
10590	0.999287	0.999014	1.00002
10680	0.999307	0.999008	1.00002
10770	0.999337	0.999000	1.00001
10860	0.999359	0.998991	1.00001
10950	0.999392	0.998989	1.00001
11040	0.999424	0.998985	1.00001
11130	0.999464	0.998976	1.00001
11220	0.999496	0.998972	1.00001
11310	0.999538	0.998966	1.00001
11400	0.999572	0.998962	1.00000
11490	0.999615	0.998958	1.00001
11580	0.999653	0.998955	1.00000
11670	0.999689	0.998952	1.00001
11760	0.999728	0.998953	1.00000
11850	0.999758	0.998949	1.00000
11940	0.999790	0.998949	1.00001
12030	0.999817	0.998948	1.00000
12120	0.999840	0.998951	1.00000
12210	0.999864	0.998955	1.00000
12300	0.999889	0.998955	1.00000
12390	0.999901	0.998961	1.00000
12480	0.999920	0.998965	1.00001
12570	0.999933	0.998976	1.00000
12660	0.999939	0.998982	1.00000
12750	0.999952	0.998994	1.00000

12840	0.999959	0.999005	1.00000
12930	0.999973	0.999022	1.00000
13020	0.999973	0.999035	1.00000
13110	0.999973	0.999061	1.00000
13200	0.999986	0.999072	1.00000
13290	0.999986	0.999094	1.00000
13380	0.999986	0.999124	1.00000
13470	0.999986	0.999146	1.00000
13560	0.999993	0.999173	1.00000
13650	0.999993	0.999203	1.00000
13740	0.999993	0.999234	1.00000

Convergence rate for y-momentum equation

4920	0.970123	0.965200	0.978607
5010	0.995888	0.994602	0.999324
5100	0.997437	0.997087	0.998064
5190	0.997606	0.997322	0.998498
5280	0.997724	0.997539	0.997928
5370	0.997680	0.997701	0.997031
5460	0.997625	0.998096	0.997155
5550	0.997923	0.998282	0.997028
5640	0.998516	0.998390	0.998863
5730	0.998964	0.998580	0.999957
5820	0.999118	0.998659	0.999464
5910	0.999066	0.999229	0.999130
6000	0.998965	0.999124	0.998995
6090	0.998887	0.998869	0.999575
6180	0.998883	0.998669	0.999523
6270	0.998953	0.998612	0.999511
6360	0.999065	0.998896	0.999656
6450	0.999186	0.999126	0.999784
6540	0.999278	0.999224	0.999909
6630	0.999347	0.999348	1.00001
6720	0.999392	0.999403	1.00007
6810	0.999428	0.999419	1.00009
6900	0.999456	0.999493	1.00009
6990	0.999481	0.999524	1.00007
7080	0.999504	0.999535	1.00004
7170	0.999524	0.999557	1.00001
7260	0.999537	0.999619	0.999992
7350	0.999542	0.999614	0.999972
7440	0.999549	0.999619	0.999960
7530	0.999554	0.999606	0.999949
7620	0.999553	0.999581	0.999943
7710	0.999553	0.999575	0.999939
7800	0.999558	0.999558	0.999934
7890	0.999560	0.999568	0.999930
7980	0.999563	0.999555	0.999923
8070	0.999562	0.999537	0.999918
8160	0.999563	0.999528	0.999913
8250	0.999564	0.999518	0.999908
8340	0.999556	0.999507	0.999907
8430	0.999558	0.999493	0.999907
8520	0.999555	0.999483	0.999906
8610	0.999548	0.999482	0.999910
8700	0.999546	0.999478	0.999911
8790	0.999545	0.999472	0.999919
8880	0.999551	0.999465	0.999923
8970	0.999545	0.999475	0.999930
9060	0.999553	0.999468	0.999934
9150	0.999562	0.999456	0.999943
9240	0.999574	0.999448	0.999947
9330	0.999580	0.999436	0.999954
9420	0.999595	0.999424	0.999960
9510	0.999612	0.999413	0.999963

9600	0.999632	0.999406	0.999970
9690	0.999646	0.999396	0.999974
9780	0.999671	0.999386	0.999976
9870	0.999679	0.999380	0.999981
9960	0.999708	0.999368	0.999983
10050	0.999730	0.999367	0.999986
10140	0.999753	0.999356	0.999988
10230	0.999768	0.999358	0.999991
10320	0.999783	0.999341	0.999993
10410	0.999800	0.999341	0.999995
10500	0.999829	0.999335	0.999995
10590	0.999837	0.999335	0.999998
10680	0.999857	0.999326	1.00000
10770	0.999867	0.999318	0.999998
10860	0.999882	0.999318	1.00000
10950	0.999895	0.999321	1.00000
11040	0.999906	0.999312	1.00000
11130	0.999916	0.999313	1.00000
11220	0.999925	0.999308	1.00000
11310	0.999934	0.999305	1.00000
11400	0.999942	0.999314	1.00000
11490	0.999949	0.999308	1.00000
11580	0.999955	0.999314	1.00000
11670	0.999961	0.999314	1.00000
11760	0.999965	0.999316	1.00000
11850	0.999970	0.999323	1.00000
11940	0.999975	0.999329	1.00000
12030	0.999977	0.999338	1.00000
12120	0.999981	0.999344	1.00000
12210	0.999983	0.999355	1.00000
12300	0.999985	0.999366	1.00000
12390	0.999987	0.999377	1.00000
12480	0.999989	0.999391	1.00000
12570	0.999991	0.999405	1.00000
12660	0.999991	0.999419	1.00000
12750	0.999994	0.999438	1.00000
12840	0.999994	0.999451	1.00000
12930	0.999995	0.999474	1.00000
13020	0.999996	0.999491	1.00000
13110	0.999996	0.999511	1.00000
13200	0.999996	0.999530	1.00000
13290	0.999998	0.999553	1.00000
13380	0.999998	0.999573	1.00000
13470	0.999999	0.999596	1.00000
13560	0.999999	0.999616	1.00000
13650	0.999999	0.999635	1.00000
13740	0.999999	0.999661	1.00000

SAJBEN TRANSONIC FLOW (Run 3)
STANDARD GRID 81 X 51

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
6030	0.985743	0.984422	0.982757
6070	0.984403	0.985798	0.989113
6110	0.980027	0.985323	0.979778
6150	0.973454	0.983269	0.975572
6190	0.992457	0.992565	0.992027
6230	1.00645	0.999919	1.01387
6270	0.996170	0.995399	0.994827
6310	0.996168	0.997734	0.998518
6350	0.999446	0.996632	1.00568
6390	0.996642	0.999289	0.996471
6430	1.00946	1.00541	1.00170
6470	0.996025	0.997029	1.00264
6510	0.996241	0.999202	0.991025
6550	0.994243	0.993600	0.992635
6590	1.00347	1.00258	1.00043
6630	1.00555	1.00045	1.01686
6670	0.993122	0.994753	0.986652
6710	0.995332	0.997740	0.996011
6750	0.994232	0.997941	0.997748
6790	1.00756	1.00105	1.01156
6830	0.993752	0.995710	0.981274
6870	1.00285	1.00350	1.00696
6910	1.00420	1.00461	1.00068
6950	0.994591	0.995620	0.994427
6990	1.00260	1.00034	0.999311
7030	0.999117	1.00215	1.00425
7070	1.00134	0.997769	1.00035
7110	1.00039	1.00163	1.00380
7150	1.00610	0.999779	1.02187
7190	0.989784	0.996395	0.977710
7230	1.00422	1.00189	1.00077
7270	1.00341	1.00168	1.01023
7310	0.999056	0.997892	0.995852
7350	0.999124	0.998770	0.994532
7390	0.994372	0.997889	0.995007
7430	1.00502	1.00059	1.01367
7470	1.00085	1.00216	0.991661
7510	1.00031	1.00384	0.991195
7550	0.994094	0.994868	0.998516
7590	1.00272	1.00178	1.00254
7630	1.00149	1.00014	1.00259
7670	1.00658	1.00436	1.00751
7710	0.991551	0.992429	0.995099
7750	1.00515	1.00611	1.00233
7790	0.993352	0.992821	0.994633
7830	1.00008	1.00057	0.996420
7870	1.00169	1.00106	1.01255
7910	0.999704	0.999549	0.983464
7950	0.999680	1.00073	1.00971
7990	0.996561	0.998170	1.00271
8030	0.998771	0.996851	0.995270
8070	1.00759	1.00695	1.00133
8110	0.986090	0.990434	0.990907
8150	1.01278	1.00803	1.00213
8190	0.991594	0.992517	0.998553
8230	1.00747	1.00680	1.00978
8270	1.00209	1.00246	0.995404
8310	0.998930	0.998674	1.00327

8350	0.993651	0.993527	0.998810
8390	1.00369	1.00328	0.997590
8430	0.995785	0.998556	0.999398
8470	1.00346	1.00228	1.00201
8510	0.996599	0.998380	0.995101
8550	1.00066	0.999763	1.00048
8590	1.00597	1.00181	1.01406
8630	0.993968	0.999474	0.983002
8670	1.00479	1.00311	1.00888
8710	0.997452	0.994657	0.997724
8750	0.996044	0.997189	0.997283
8790	0.988574	0.990444	0.991853
8830	1.01341	1.00904	1.01257
8870	1.00024	1.00296	0.997771
8910	0.999492	0.997947	0.994411
8950	0.998075	0.997050	1.01135
8990	1.00702	1.00988	1.00163

Convergence rate for x-momentum equation

6030	0.980510	0.986001	0.981736
6070	0.983497	0.987881	0.985533
6110	0.988484	0.991909	0.986713
6150	0.990513	0.993054	0.980786
6190	0.996786	0.996099	1.00491
6230	1.00460	1.00357	0.995886
6270	0.997216	0.996475	1.00547
6310	1.00551	0.996861	1.03694
6350	1.00276	0.997867	0.992839
6390	0.987441	0.993958	0.978793
6430	1.00952	1.00363	1.00617
6470	0.999057	0.999055	1.00330
6510	0.997018	1.00017	0.990210
6550	0.988754	0.989808	0.996650
6590	1.00708	1.00445	1.00476
6630	1.00460	1.00153	1.00585
6670	1.00633	0.999051	1.01498
6710	0.983586	0.994787	0.970823
6750	0.996540	0.996780	1.00189
6790	1.00769	1.00163	1.00630
6830	0.996525	0.996998	0.994782
6870	0.999992	1.00213	0.998241
6910	1.00344	1.00236	0.996804
6950	0.994506	0.996123	0.998797
6990	1.00153	1.00124	0.998053
7030	1.00504	1.00531	1.00662
7070	1.00120	0.997008	1.00528
7110	1.00845	1.00501	1.01620
7150	0.996781	0.997976	1.00075
7190	0.995242	0.997235	0.995667
7230	0.997500	1.00022	0.983668
7270	0.999631	0.999194	1.00485
7310	1.00209	1.00081	1.00231
7350	1.00533	1.00068	1.01035
7390	0.990607	0.995739	0.981506
7430	1.00457	1.00046	1.01863
7470	0.996763	1.00140	0.979126
7510	1.00544	1.00539	1.00928
7550	0.991487	0.993097	0.990086
7590	1.00420	1.00331	1.00646
7630	1.00383	1.00145	1.01003
7670	0.997657	1.00031	0.984584
7710	0.996167	0.993655	0.999941
7750	1.01076	1.00909	1.01432
7790	0.987227	0.989973	0.989908
7830	1.00151	1.00343	0.993835

7870	1.00231	0.998760	1.01234
7910	1.00129	1.00116	0.998643
7950	0.997844	0.998868	0.998938
7990	1.00028	1.00156	0.997672
8030	0.995533	0.996853	0.994099
8070	1.00413	1.00360	1.00256
8110	0.993654	0.992750	1.00265
8150	1.00705	1.00472	1.00114
8190	0.992284	0.995023	0.996953
8230	1.00891	1.00874	0.999257
8270	1.01176	1.00134	1.03486
8310	0.987412	0.996975	0.968984
8350	0.992374	0.994045	0.997439
8390	1.00921	1.00630	1.00552
8430	0.993333	0.994775	0.996116
8470	1.00154	1.00256	0.996654
8510	1.00239	1.00056	1.00947
8550	0.996850	0.996914	0.996493
8590	1.00729	1.00463	1.00984
8630	0.991121	0.996407	0.985079
8670	1.00429	1.00232	1.00263
8710	1.00555	1.00079	1.00811
8750	0.991151	0.995707	0.984591
8790	0.996291	0.992527	1.00620
8830	1.00946	1.00878	1.00591
8870	0.998626	1.00159	0.991748
8910	0.993366	0.993901	1.00589
8950	1.00636	1.00125	1.00428
8990	0.999665	1.00567	0.992499

Convergence rate for y-momentum equation

6030	0.981997	0.983112	0.990836
6070	0.981113	0.983837	0.977448
6110	0.979207	0.981370	0.979795
6150	0.980103	0.984310	0.989316
6190	0.994589	0.991689	1.00093
6230	1.00909	1.00383	1.00734
6270	1.00275	0.998961	1.00378
6310	0.994257	0.997799	0.986503
6350	1.01125	1.00659	1.01571
6390	0.989402	0.993478	0.991326
6430	1.00701	1.00530	1.00182
6470	1.00573	0.999716	1.02148
6510	0.988923	0.998833	0.972546
6550	0.998748	0.990928	1.02139
6590	0.998183	1.00481	0.978497
6630	1.00989	1.00499	1.02600
6670	1.00596	1.00169	0.997068
6710	0.984589	0.992028	0.975270
6750	0.990615	0.994667	1.00040
6790	1.00689	1.00358	1.00145
6830	1.00084	0.997388	1.00936
6870	1.00060	1.00086	1.00297
6910	1.00241	1.00584	0.983145
6950	0.996014	0.996378	1.00762
6990	1.00020	0.999330	0.995573
7030	1.00527	1.00369	1.00744
7070	1.00479	1.00189	1.01245
7110	1.00487	1.00347	1.00560
7150	0.993756	0.996972	0.985155
7190	0.999711	0.996084	1.00962
7230	0.994623	0.999624	0.986413
7270	1.00480	1.00369	1.00225
7310	0.994020	0.994932	1.00015
7350	1.00847	1.00375	0.998336

7390	0.989486	0.995025	0.992447
7430	1.00697	1.00281	1.00833
7470	1.00001	0.999925	1.00187
7510	1.00152	1.00581	0.995772
7550	0.989211	0.989498	0.987905
7590	1.00339	1.00456	1.00944
7630	1.00483	1.00241	1.00143
7670	1.00559	1.00378	1.01611
7710	0.984546	0.988615	0.983122
7750	1.01180	1.01055	0.998477
7790	0.992543	0.990908	0.997060
7830	1.00032	1.00276	0.997009
7870	0.999815	0.999425	0.994076
7910	1.00200	1.00083	1.00454
7950	0.995996	0.997272	0.998836
7990	1.00038	0.999443	1.00441
8030	0.998021	0.998328	0.990800
8070	1.00364	1.00479	1.01025
8110	0.988467	0.989806	0.992857
8150	1.01106	1.00822	1.01508
8190	0.987176	0.990284	0.969283
8230	1.01099	1.01079	1.01699
8270	1.01181	1.00314	1.01782
8310	0.987328	0.994441	0.981805
8350	0.990992	0.993675	0.988086
8390	1.01583	1.01004	1.02229
8430	0.994199	0.997371	0.997398
8470	1.00256	1.00126	0.998523
8510	0.996410	0.997128	0.994748
8550	0.999076	0.999709	0.999508
8590	1.00502	1.00558	1.00386
8630	0.992724	0.995327	0.989006
8670	1.00389	1.00211	1.01398
8710	1.00285	0.998726	0.994382
8750	0.995223	0.995660	1.00015
8790	0.991187	0.990595	0.997447
8830	1.00600	1.00510	1.01221
8870	0.996158	1.00115	0.977333
8910	1.00323	1.00050	1.00558
8950	1.00592	1.00209	1.01774
8990	0.997980	1.00485	0.986659

SAJBEN TRANSONIC FLOW (Run 3)
TWICE AS COARSE

iter	resl2	resavg	resmax

Convergence rate for continuity equation

6030	0.974489	0.977890	0.977506
6070	0.991778	0.990260	0.979107
6110	0.990214	0.990157	0.995800
6150	0.995725	0.993539	1.00181
6190	1.00154	1.00170	0.997744
6230	0.997876	0.996993	0.999128
6270	1.00565	1.00776	1.01034
6310	0.992516	0.991075	0.985698
6350	0.984208	0.985748	0.987518
6390	1.00011	1.00285	0.991677
6430	1.00826	1.00294	1.02864
6470	0.999091	0.997360	0.999738
6510	0.994972	0.998377	0.990677
6550	1.01321	1.00993	1.01135
6590	0.997336	0.997818	1.00016
6630	0.998434	0.998909	0.996175
6670	0.994762	0.990908	0.995735
6710	0.992721	0.998018	0.984886
6750	1.00991	1.00688	1.01360
6790	0.999992	0.999863	1.00795
6830	0.986045	0.987851	0.985397
6870	0.997605	0.996167	0.991420
6910	1.00542	1.00105	1.01651
6950	1.00890	1.00738	1.01030
6990	1.00749	1.01267	1.00482
7030	0.996093	1.00049	0.980105
7070	0.993589	0.992749	0.997015
7110	0.997863	0.992933	1.00722
7150	1.00088	0.999085	0.996247
7190	1.00784	1.00666	1.01951
7230	0.993845	0.997715	0.986697
7270	0.997927	0.997954	1.00063
7310	0.990559	0.990605	0.989286
7350	1.00529	1.00792	1.00407
7390	1.00894	1.00697	0.999435
7430	0.991582	0.993445	0.997825
7470	1.00678	1.00021	1.01588
7510	0.993633	0.992433	0.991196
7550	1.00654	1.01364	0.996482
7590	0.995445	0.995636	0.996451
7630	0.994661	0.992182	0.992729
7670	1.01133	1.01167	1.00978
7710	0.992054	0.992395	1.00852
7750	1.00216	0.998122	0.999292
7790	1.00067	0.996227	1.00901
7830	1.00950	1.01921	0.988491
7870	1.00126	0.993993	1.01360
7910	0.980813	0.988125	0.975630
7950	1.01383	1.00997	1.01995
7990	0.993723	0.997598	0.984940
8030	1.00873	1.00572	1.01030
8070	0.995232	0.997967	0.989385
8110	1.00252	1.00260	1.00778
8150	0.999277	0.995535	1.00418
8190	1.00541	1.00427	1.01126
8230	0.992783	0.995534	0.986265
8270	0.996244	0.997512	0.991811
8310	0.996006	0.995723	1.00117

8350	1.00637	1.00540	1.00236
8390	1.00615	1.00450	1.01404
8430	1.00317	1.00269	0.995701
8470	0.996431	0.998552	1.00271
8510	0.998162	0.998294	0.996507
8550	0.993933	0.995867	0.981537
8590	0.994471	0.995250	0.995265
8630	1.01503	1.01112	1.02893
8670	0.998956	0.998543	0.996766
8710	0.991202	0.987007	0.997634
8750	0.991950	0.999719	0.985152
8790	1.01755	1.01405	1.01476
8830	0.988122	0.992627	0.993429
8870	0.996188	0.990827	0.991834
8910	1.01112	1.00260	1.02730
8950	1.01738	1.02728	1.00032
8990	0.987700	0.985325	0.993512

Convergence rate for x-momentum equation

6030	0.983979	0.984598	0.996997
6070	1.00058	0.996694	1.01493
6110	0.995552	0.994613	1.00007
6150	0.993455	0.988097	0.997392
6190	0.996104	0.999310	0.985324
6230	1.00063	0.998651	1.00264
6270	1.01302	1.01171	1.01973
6310	0.990170	0.994604	0.977076
6350	0.984582	0.982370	1.00060
6390	1.00668	1.00523	0.995230
6430	1.00836	1.00266	1.01189
6470	1.00131	1.00133	0.999736
6510	0.995026	0.995734	1.00065
6550	1.00432	1.00626	0.992215
6590	1.00177	1.00248	1.01196
6630	1.00048	0.999103	0.997622
6670	0.999647	0.995823	1.00067
6710	0.995490	0.997854	0.987081
6750	1.00217	1.00240	1.01258
6790	1.00381	1.00251	1.00448
6830	0.990069	0.989680	0.991997
6870	0.995650	0.995342	0.996199
6910	1.00560	1.00476	1.00476
6950	1.00051	1.00488	0.996082
6990	1.00886	1.00847	1.01663
7030	0.997291	1.00157	0.990926
7070	0.992555	0.993372	0.986370
7110	1.00360	0.997619	1.00853
7150	1.00297	0.999093	1.00297
7190	0.999699	1.00234	1.00048
7230	0.998050	0.999144	0.996576
7270	0.994551	0.995881	0.997335
7310	1.00026	0.998687	0.992858
7350	0.991431	0.997717	0.995134
7390	1.01393	1.00601	1.02825
7430	0.991503	0.996886	0.973700
7470	1.00492	1.00022	1.02341
7510	1.00113	0.994524	0.993721
7550	1.00665	1.01257	1.00001
7590	0.995349	0.997351	1.00539
7630	0.988971	0.987929	0.984843
7670	1.02570	1.01578	1.03192
7710	1.01079	0.999960	1.01620
7750	0.967672	0.985288	0.951779
7790	1.00390	0.999111	1.01717
7830	1.01174	1.02050	1.00069

7870	0.998081	0.993246	1.00399
7910	0.989932	0.994114	0.982724
7950	1.00490	1.00597	1.00325
7990	0.995617	0.997491	0.991850
8030	1.00312	0.998788	1.01051
8070	0.999232	0.999915	0.998389
8110	1.00497	1.00501	1.00432
8150	0.995313	0.996818	0.992738
8190	1.00677	1.00612	1.01064
8230	0.997956	0.996052	1.00045
8270	0.992692	0.996035	0.994922
8310	0.994955	0.994954	0.985672
8350	1.00415	1.00255	1.01076
8390	1.00848	1.00673	1.00531
8430	0.992895	0.996273	0.996491
8470	1.01540	1.00330	1.02283
8510	0.987570	0.998345	0.976217
8550	0.997319	0.998612	0.996207
8590	1.00248	0.997907	1.01073
8630	1.00257	1.00551	0.989521
8670	0.996974	0.994289	1.01200
8710	0.997987	0.993822	0.992691
8750	0.999882	1.00139	1.00544
8790	1.00105	1.00639	0.997494
8830	0.996947	0.995741	1.00146
8870	0.997983	0.994198	0.992861
8910	1.00820	1.00846	1.01293
8950	1.00699	1.01523	0.995883
8990	0.995970	0.990210	1.00328

Convergence rate for y-momentum equation

6030	0.960685	0.965498	0.974013
6070	1.01031	0.998549	1.02137
6110	1.00267	1.00020	1.01550
6150	0.995047	0.988029	0.996988
6190	0.988260	0.996296	0.974302
6230	1.00679	1.00534	1.01269
6270	1.00860	1.01326	0.997969
6310	0.992222	0.992091	1.00110
6350	0.980929	0.979708	0.981117
6390	1.01172	1.00963	1.00727
6430	1.00810	1.00885	1.00703
6470	1.00286	1.00008	1.00514
6510	0.993198	0.994814	0.996843
6550	1.00670	1.00766	1.00828
6590	0.999847	1.00018	0.995630
6630	1.00164	1.00194	1.00196
6670	0.999613	0.993762	0.995755
6710	0.986546	0.992742	0.982435
6750	1.01623	1.01242	1.02450
6790	0.995707	0.995530	0.994676
6830	0.998416	0.994274	1.00024
6870	0.993754	0.994795	0.995340
6910	0.998057	0.998982	0.999611
6950	1.00719	1.00751	1.00934
6990	1.00257	1.00928	0.999701
7030	1.00053	1.00189	0.989381
7070	0.992395	0.991072	0.996112
7110	1.00934	1.00237	1.01331
7150	0.984780	0.989059	0.989263
7190	1.00885	1.00908	0.996251
7230	0.995335	0.997193	0.999004
7270	1.00826	0.999672	1.01866
7310	0.997333	0.997273	0.995445
7350	0.995133	0.999372	0.991994

7390	1.00287	1.00605	1.00578
7430	1.00252	0.996439	0.996880
7470	0.999683	1.00071	1.00094
7510	1.00170	0.996451	1.00768
7550	1.00249	1.01166	0.996642
7590	1.00100	0.997820	1.00388
7630	0.982629	0.984399	0.978296
7670	1.01464	1.01662	1.01960
7710	0.999440	0.996727	0.993143
7750	0.999023	0.997530	0.998525
7790	0.997819	0.995090	0.999544
7830	1.00553	1.01277	1.00998
7870	1.00381	1.00028	0.997147
7910	0.978732	0.984354	0.978405
7950	1.01563	1.00876	1.02478
7990	0.996921	0.997898	0.997153
8030	1.00744	1.00932	1.00625
8070	0.996254	0.994250	0.993216
8110	0.998210	0.998901	1.00384
8150	1.00267	1.00228	0.999663
8190	0.995597	0.997889	0.994377
8230	1.00402	1.00221	1.00886
8270	0.994555	0.995013	0.987922
8310	1.00326	0.999035	1.00414
8350	0.997354	1.00161	0.994868
8390	1.00429	1.00456	1.00348
8430	1.00597	1.00458	1.01427
8470	0.993122	0.997088	0.990641
8510	1.00261	1.00215	0.998453
8550	0.994888	0.992488	0.998642
8590	1.00346	0.999720	1.00536
8630	1.00039	1.00632	0.990778
8670	0.997694	0.995707	1.00111
8710	1.00332	0.996281	1.00930
8750	0.997682	0.997936	0.999007
8790	1.00158	1.00969	0.992349
8830	1.00153	0.996539	1.00429
8870	0.991559	0.990869	0.990657
8910	0.998238	1.00034	1.00319
8950	1.01608	1.02102	1.00675
8990	0.992727	0.990375	1.00108

SAJBEN TRANSONIC FLOW (Run 3)
TWICE AS FINE

iter	resl2	resavg	resmax
Convergence rate for continuity equation			
6030	1.00050	1.00216	0.988710
6070	0.997701	1.00021	0.997385
6110	0.994742	0.996831	1.00180
6150	0.993643	0.994346	0.995274
6190	0.992816	0.993693	0.991985
6230	0.992210	0.991719	0.991866
6270	0.992906	0.992600	1.00476
6310	0.991305	0.992011	0.986801
6350	0.992751	0.992266	0.998828
6390	0.992521	0.993811	0.988048
6430	0.990441	0.993017	0.997519
6470	0.988566	0.993136	0.990607
6510	0.984873	0.991333	0.980649
6550	0.995892	0.995359	1.00220
6590	0.997920	0.996542	0.995923
6630	1.00070	0.998115	0.995177
6670	1.00437	1.00069	1.01899
6710	0.999478	0.999795	0.989736
6750	1.00676	1.00153	1.01141
6790	0.995810	0.996457	1.00143
6830	0.996787	0.999550	0.982836
6870	1.00139	0.998175	1.01398
6910	0.995279	0.998207	0.989238
6950	1.00643	0.998744	1.01478
6990	1.00195	1.00225	0.997338
7030	0.994701	0.997689	1.00101
7070	1.00772	1.00046	1.01303
7110	0.988474	0.996941	0.978319
7150	1.00493	1.00094	1.00228
7190	0.992233	0.996226	0.995262
7230	1.00070	0.997553	1.00207
7270	1.00363	0.998591	1.00731
7310	0.998896	1.00022	0.986487
7350	1.00138	0.997324	1.01062
7390	1.00253	1.00297	1.00394
7430	1.00949	1.00550	1.01660
7470	0.994233	0.998172	0.983401
7510	0.988435	0.993310	0.989654
7550	1.00111	0.998126	1.00089
7590	1.01085	1.00792	1.01859
7630	0.994861	0.997266	0.985498
7670	0.998760	0.999869	0.998626
7710	0.998856	0.999016	1.00307
7750	1.00971	1.00644	1.00548
7790	1.00253	0.999384	1.00742
7830	0.998371	0.998064	0.993720
7870	0.990073	0.996713	0.995759
7910	1.00302	1.00311	0.998013
7950	1.00639	1.00172	1.00739
7990	1.00462	1.00125	1.00394
8030	0.988912	0.996241	0.985094
8070	1.01035	1.00456	1.01222
8110	0.991474	0.994823	0.999483
8150	0.997965	1.00062	0.990391
8190	1.02006	1.00961	1.02738
8230	0.976907	0.989606	0.967648
8270	1.00333	0.995443	1.01653
8310	0.993610	0.997764	0.981728

8350	1.00303	0.997739	1.00894
8390	1.00104	1.00164	0.998565
8430	0.997280	1.00196	1.00480
8470	0.996215	0.997259	0.988904
8510	1.00733	1.00504	1.01069
8550	0.999767	0.994410	0.997051
8590	0.998678	1.00367	0.995965
8630	0.999689	0.998915	0.999155
8670	1.00036	0.998021	1.00417
8710	1.00402	1.00441	1.00641
8750	0.998961	0.999694	0.993266
8790	1.00288	1.00140	1.01204
8830	1.00044	1.00328	0.992496
8870	0.998200	0.996725	0.999855
8910	0.996930	0.999350	0.994738
8950	1.00066	1.00208	1.00120
8990	0.999692	0.997219	0.998300

Convergence rate for x-momentum equation

6030	0.999523	1.00064	1.01081
6070	0.995548	0.999851	0.987751
6110	1.00371	1.00053	1.02020
6150	0.998422	0.996712	0.994695
6190	0.982669	0.991137	0.980716
6230	0.996731	0.994478	0.998924
6270	0.992945	0.994032	0.988604
6310	0.988994	0.992178	0.993454
6350	1.00454	0.996628	1.01053
6390	0.989625	0.994435	0.990564
6430	0.991738	0.995855	0.980047
6470	0.995699	0.995619	1.00749
6510	0.987920	0.994717	0.970679
6550	1.00258	0.999244	1.01798
6590	0.996150	0.997840	0.992983
6630	0.993689	0.996254	0.994644
6670	1.00234	1.00026	0.996305
6710	0.999136	0.998959	0.997781
6750	1.01608	1.00289	1.02806
6790	0.993845	0.997226	0.985916
6830	0.995255	0.997581	0.999604
6870	1.00871	1.00077	1.01570
6910	0.991215	0.997790	0.981343
6950	1.00969	0.998932	1.01395
6990	0.998412	1.00207	0.994825
7030	0.996119	0.996814	0.998435
7070	1.01176	1.00349	1.01546
7110	0.988820	0.996406	0.983213
7150	1.00685	1.00165	1.00576
7190	0.993681	0.996484	0.997388
7230	0.998607	0.993905	1.00387
7270	0.999549	0.998882	1.00102
7310	1.00259	1.00097	0.998933
7350	0.999763	0.995501	1.00627
7390	1.00560	1.00587	0.994992
7430	1.00322	1.00299	1.01578
7470	1.00238	0.999794	1.00125
7510	0.985563	0.993111	0.974799
7550	1.00011	0.997261	1.01141
7590	1.01410	1.00992	1.00981
7630	0.992472	0.995974	0.989716
7670	0.996908	0.999696	0.991078
7710	1.00354	1.00194	1.00684
7750	1.00357	1.00331	1.00408
7790	1.00102	0.998403	0.996054
7830	1.00314	1.00129	1.00770

7870	1.00427	1.00227	1.00540
7910	0.993795	0.997800	0.991638
7950	1.00773	1.00662	1.00803
7990	0.997739	0.997264	1.00266
8030	0.993244	0.997156	0.992484
8070	1.00888	1.00532	1.00115
8110	0.986864	0.992432	0.993049
8150	0.996382	0.999224	0.988039
8190	1.02408	1.01318	1.03301
8230	0.979007	0.988269	0.978261
8270	1.00095	0.993080	1.01467
8310	0.991061	0.995668	0.986317
8350	1.00259	1.00064	0.997411
8390	1.00627	1.00405	1.00728
8430	0.999219	1.00013	1.00164
8470	0.995890	0.997101	0.987844
8510	1.00561	1.00547	1.00563
8550	0.998651	0.996977	0.997744
8590	0.999570	0.999553	0.998992
8630	1.00156	1.00092	1.00782
8670	0.998703	0.999306	0.994219
8710	1.00099	1.00174	1.00664
8750	1.00385	1.00293	1.00210
8790	0.995162	0.996361	0.990226
8830	1.00036	1.00419	0.998776
8870	1.00427	0.998373	1.00936
8910	0.996481	1.00041	1.00014
8950	1.00273	1.00193	1.00757
8990	0.994597	0.996255	0.981169

Convergence rate for y-momentum equation

6030	1.00518	1.00606	1.01421
6070	0.995899	0.998777	0.980055
6110	0.996079	0.997059	1.01047
6150	0.995805	0.994419	1.01235
6190	0.985652	0.989883	0.975980
6230	0.991335	0.990196	0.991237
6270	0.989543	0.992170	0.984778
6310	0.990938	0.988457	1.00626
6350	0.993870	0.992901	0.985806
6390	0.991908	0.991403	1.00430
6430	0.991818	0.994206	0.985965
6470	0.993035	0.992949	0.986529
6510	0.994289	0.994365	1.00028
6550	1.00133	0.998178	0.992956
6590	0.994542	0.997065	0.991760
6630	1.00170	0.998421	1.00345
6670	1.00305	1.00207	1.00830
6710	0.996047	0.998754	0.991900
6750	1.00542	1.00308	1.00226
6790	1.00200	1.00092	1.00264
6830	0.996315	0.996188	1.00393
6870	1.00860	1.00417	1.00946
6910	0.991750	0.995168	0.991042
6950	1.00794	1.00322	1.01516
6990	0.999313	1.00316	0.983334
7030	0.995547	0.996535	0.997551
7070	1.00749	1.00164	1.02129
7110	0.993345	0.999421	0.977152
7150	1.00600	1.00363	1.01097
7190	0.992977	0.993600	0.989103
7230	0.996247	0.993602	1.00608
7270	1.00216	0.999170	1.00784
7310	0.996250	0.999794	0.988096
7350	1.00269	0.999192	1.00679

7390	1.00431	1.00486	0.997153
7430	1.00495	1.00494	1.01486
7470	0.999911	0.996774	1.00039
7510	0.997028	1.00023	0.985291
7550	0.998719	0.995541	1.00052
7590	1.00771	1.00837	1.01732
7630	0.995604	0.997038	0.989307
7670	0.997536	0.996999	0.996034
7710	1.00510	1.00355	1.02171
7750	0.996618	1.00349	0.976667
7790	0.998388	0.995916	0.998754
7830	1.00430	1.00146	1.01234
7870	1.00215	1.00159	1.00260
7910	0.999227	1.00038	1.00506
7950	0.999407	1.00085	0.990957
7990	0.994885	0.996996	0.993157
8030	1.00498	1.00247	1.00392
8070	1.00812	1.00438	1.01522
8110	0.986443	0.994122	0.981095
8150	0.998668	0.997712	0.987531
8190	1.01248	1.00955	1.02158
8230	0.988698	0.991261	0.993363
8270	0.997360	0.994962	1.00014
8310	0.998242	0.996017	1.00190
8350	1.00033	1.00098	0.988300
8390	1.00771	1.00663	1.01800
8430	0.995071	0.997202	0.987369
8470	1.00116	0.999874	1.00285
8510	1.00151	1.00265	1.00525
8550	1.00093	0.997351	1.00396
8590	0.993123	0.996534	0.999484
8630	1.00774	1.00488	1.00203
8670	0.998623	1.00028	0.991959
8710	0.995628	0.998243	0.994745
8750	1.00840	1.00569	1.00911
8790	0.994581	0.996410	0.991826
8830	0.999680	1.00228	1.00007
8870	1.00237	0.998177	1.00451
8910	0.998714	1.00052	0.996824
8950	1.00333	1.00309	1.01637
8990	0.992334	0.993935	0.976076